Restoration of Finger Movement using Functional Electrical Stimulation and Bayes' Theorem

A Thesis
Presented to

The Biomedical Engineering
Interdisciplinary Program
of the University of Arizona

in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

In Biomedical Engineering

November 2001

Committee Chair: Andrew Fuglevand

TABLE OF CONTENTS

DATE: October 23, 2001

TITLE: Restoration of Movement using Functional Electrical Stimulation and Bayes' Theorem

AUTHORS: Heather M. Seifert and Andrew J. Fuglevand

INSTITUTION: Department of Physiology and Program in Biomedical Engineering, University of Arizona, Tucson, USA

CORRESPONDENCE: Andrew J. Fuglevand, Ph.D.
Department of Physiology
College of Medicine
P.O. Box 210093
University of Arizona
Tucson, AZ 85721-0093
Phone: (520) 621-6983
Fax: (520) 621-8170
Email: fuglevan@u.arizona.edu

ACKNOWLEDGEMENTS:

We gratefully acknowledge the advice and encouragement from Richard Zemel and Bruce McNaughton and the technical assistance of Claudia Stanescu. This work was supported by grant NIH NS 39489 (to AJF).

ABSTRACT

Various computational approaches have been applied to predict aspects of animal behavior from the recorded activity of populations of neurons. Here we invert this process to predict the requisite neuromuscular activity associated with specified motor behaviors. A probabilistic method based on Bayes' theorem was used to predict the patterns of muscular activity needed to produce various types of desired finger movements. The profiles of predicted activity were then used to drive frequency-modulated muscle stimulators in order to evoke multi-joint finger movements. Comparison of movements generated by electrical stimulation to desired movements yielded root mean squared errors between ~18 – 26%. This reasonable correspondence between desired and evoked movements suggests that this approach might serve as a useful strategy to control neuroprosthetic systems that aim to restore movement to paralyzed individuals.

INTRODUCTION

Fundamental insights into how arrays of neurons encode motor or sensory variables can be gained from computational methods that attempt to reconstruct or predict aspects of animal behavior or sensory stimuli from the recorded activity of neural populations (Georgopoulos et al. 1986, 1988; Schwartz, 1993;Wilson and McNaughton 1993; Deadwyler and Hampson, 1997; Rieke et al. 1997 Brown et al. 1998; Nicolelis et al. 1998; Zhang et al. 1998; Wessberg et al. 2001). The accuracy with which a behavior such as the direction of limb movement or the path of an animal navigating a maze can be reconstructed, provides an estimate of the amount of behaviorally-relevant information represented in the discharge of the recorded neurons.

It should also be possible to invert this process in order to predict neural activity from behavior. One application of such an approach would be to identify the patterns of neuromuscular activity across a population of muscles needed to elicit desired movements in paralyzed individuals using functional electrical stimulation. Functional electrical stimulation involves artificial activation of paralyzed muscles with implanted electrodes (Keith et al. 1988; Hoshimiya et al. 1989; Kilgore et al. 1989; Smith et al. 1998) and has been successfully used to improve the ability of quadriplegics to perform activities for daily living (Mulcahey et al. 1997). The range of motor behaviors that can be generated by functional electrical stimulation, however, is limited to a relatively small set of preprogrammed movements such as hand grasp, lateral and palmar pinch (Triolo et al. 1996).

In an attempt to overcome this limitation, we have used a probabilistic method called Bayes' theorem to predict the patterns of muscle stimulation needed to produce, in

theory, an unlimited set of movements across multiple joints. Our use of Bayes' theorem was based on previous studies that used this method to reconstruct various forms of motor behavior from recorded neural activity (Brown et al. 1998; Zhang et al. 1998; Tresch and Kiehn, 2000). The bidirectionality of Bayes' theorem facilitated the inverse prediction of neuromuscular activity from behavior required for the present investigation (Rieke et al. 1997). The aim of this study, therefore, was to determine if implementation of Bayes' theorem was an effective method for predicting muscle stimulation patterns needed to artificially evoke a variety of finger movements. A reasonable correspondence between desired and evoked movements was observed in this study, indicating that this approach might provide a flexible means to control functional electrical stimulation and thereby expand the repertoire of motor functions available to paralyzed individuals. An abstract of this work has been published (Seifert et al. 2001).

MATERIALS AND METHODS

Overview

The general approach taken in this study involved two stages as outlined in Figure 1. In the first stage, electromyographic (EMG) and joint kinematic signals were recorded during a variety of finger movements in one subject. These signals were then used as inputs to a computer algorithm that characterized the relationship between muscle activity and kinematics using a probabilistic method known as Bayes' theorem. In the second stage, the probabilistic relationship between muscle activity and kinematics identified in the first stage was used to predict muscle activity associated with a new set of intended or desired movements of the finger. The predicted patterns of muscle activity were then transformed into frequency-modulated trains of pulses that were used to control a set of muscle stimulators in order to evoke finger movements in other subjects. The accuracy of the method was evaluated by comparing evoked movements to the corresponding desired movements. Details of the procedures are given in the following sections. The Institutional Human Investigation Committee approved the procedures and all subjects gave their informed consent to participate in the study.

Insert FIGURE 1 about here

Joint Angle and EMG Acquisition

A healthy human subject sat in a dental chair with the forearm supported on a platform and stabilized in a mid-supinated position between two foam-padded rods as shown in Figure 2. Three flexible strain gauge transducers (Biopac Inc. USA) were used to record joint angles from the metacarpalphalangal joint (MCP), the proximal interphalangeal joint (PIP), and the distal interphlangeal (DIP) of the third digit. This

digit was used because fewer muscles insert onto it compared to the thumb, index finger, or little finger and because of its greater independence of movement compared to digit four (Robinson and Fuglevand, 1998; Häger-Ross and Schieber, 2000). The joint angle transducers were attached with double-sided tape across each joint after the subject had donned a vinyl glove. The glove was worn to improve adhesion of the transducers. A plastic extension was glued to the glove over the finger nail in order to lengthen the distal segment and thereby allow the transducer to be fixed across the DIP joint. Once attached to the subject, each of the joint angle transducers was calibrated using a metal frame that held the joints at specified angles. Angular position was measured with respect to a neutral (fully extended) orientation of the joints with positive angles referring to flexion and negative angles indicating hyperextension. Joint angle signals were amplified (gain of 1000, World Precision Instruments USA) and sampled with a computerized data acquisition system (Spike 2, Cambridge Electronics Design UK) at ~ 2000 Hz.

Insert FIGURE 2 about here

Tungsten microelectrodes were used to record EMG signals from the main muscles that control flexion and extension of the third digit, i.e, the digit III compartments of the flexor digitorum profundus (FDP3), flexor digitorum superficialus (FDS3), and extensor digitorum (ED3). The tungsten microelectrodes (1-5 μm tip diameter, ~ 3 mm of insulation removed from the tip, 250 μm shaft diameter, Frederick Haer Co. USA) were inserted through the skin and directed toward the target muscle. Low-intensity constant current pulses (~ 0.4 mA, 1 ms duration, 1 pulse/s) were delivered via a stimulator coupled to a stimulus isolation unit (Grass S88 and SIU7 USA) while the intramuscular electrode position was adjusted manually until a site was found that elicited

motor responses in one of the target muscles. Activation of FDP was distinguished from that of FDS by the presence of evoked movements in the distal phalanx. Once the placement of the electrodes in the target muscles had been verified by electrical stimulation, the electrodes were then connected to AC coupled differential amplifiers (Grass model 12 USA). Surface electrodes (Ag-AgCl, 4 mm diameter) attached to the skin over the distal radius served as reference electrodes. EMG signals were amplified with a gain of 1000, band pass filtered (30 – 1000 Hz), and digitally sampled at ~ 2000 Hz.

Training Data

Once the position transducers and electrodes were in place, the subject was asked to perform a variety of unrestrained flexion-extension movements of the middle finger in which contact was not made with external surfaces. Some movement of the other fingers also occurred inadvertently. EMG and joint angle data, however, were recorded only for the middle finger movements which were used for subsequent training of the Bayes' algorithm to yield the probabilistic relationships between muscle activity and joint kinematics. The movements were designed to cover much of the joint space associated with relatively natural movements. The duration of the training set was 60 s.

Desired Movements

Next, the subject was instructed to make a sequence of movements from which a set of desired movements were extracted from the recorded joint angle trajectories. These movements consisted of repeated tapping motions similar to key presses, pushing movements involving simultaneous extension of the PIP and DIP joints and flexion of the MCP joint of the middle finger in a motion like that which occurs when sliding a small

object away from the hand across a flat surface, and pulling movements involving flexion of the PIP and DIP joints and extension of the MCP joint as if sliding an small object toward the hand. The three types of movements were performed repetitively for about 10 s each. The subject was instructed to make movements at a comfortable pace but to vary the duration of the movement from one cycle to the next. The entire 30 s sequence was performed twice. From this record, five 10-s segments were extracted which were used to represent different types of desired movements: tapping, pushing, pulling, transition from pushing to tapping movements, and transition from tapping into pulling movements.

Signal Processing

In off-line digital analysis of the training set, EMG signals were full-wave rectified and low-pass filtered at 2 Hz. Joint angular velocities were calculated for each joint by digital differentiation of the joint angle data. Positive values for joint angular velocity indicated flexion movements whereas negative values indicated extension movements. Joint angle, joint angular velocity, and EMG signals were all re-sampled at ~ 200 Hz/signal. EMG magnitude was normalized to a percentage of the peak EMG within the training set and rounded to the nearest 1% increment. Joint angles and joint angular velocities were rounded into intervals of one degree and one degree/s, respectively.

Bayesian Reconstruction Algorithm

Bayes' theorem is a technique that uses conditional probabilities to predict the likelihood of an outcome given that a particular event or set of events has occurred. The basic form of Bayes' theorem can be written as:

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)} \qquad (1)$$

where P(A|B) is the probability that variable A takes on a particular value given different levels of variable B. In neurophysiology experiments, A is often a controlled parameter related to a sensory stimulus or a behavior and B typically is an index of neural activity. P(B|A) is the probability that variable B attains a specific value given different levels of A. P(A) is the distribution representing the probabilities for observing different levels of A. In practice, the denominator term P(B) is treated as a normalization constant that represents the sum of probabilities across all levels of A for the distribution indicated in the numerator of Equation 1, namely:

$$P(B) = \sum_{allA} P(B \mid A) \cdot P(A). \tag{2}$$

This normalization simply ensures that total probability represented by Equation 1 is equal to 1.0.

In the present case, the variables of interest were joint kinematics (θ) and muscle activity (EMG). In contrast to previous studies, in which neural activity has been used to predict some aspect of behavior (Georgopoulos et al. 1986; 1988; Schwartz, 1993; Wilson and McNaughton, 1993, Tresch and Kiehn, 2000) or features of sensory stimuli (Rieke et al. 1997, Nicolelis et al. 1998), our goal was predict the requisite neuromuscular activity needed to generate a particular motor behavior. Consequently, the general form of equation 1 became:

$$P(EMG \mid \Theta) = \frac{P(\Theta \mid EMG) \cdot P(EMG)}{\sum_{allEMG} P(\Theta \mid EMG) \cdot P(EMG)} \tag{3}$$

In our application of Bayes' theorem, six kinematic variables– three joint angle trajectories (Θ_j) and the three associated joint angular velocities ($\dot{\Theta}_j$) – were used to

predict activity in a muscle (EMG_i). Equation 3 was first applied individually for each of the six kinematic parameters. Then, under the simplifying assumption of independence among kinematic parameters, the probability of EMG given values for all six kinematic parameters was given by the product of the individual probabilities, namely:

$$P(EMG_i \mid \Theta_1, \Theta_2, \Theta_3, \dot{\Theta}_1, \dot{\Theta}_2, \dot{\Theta}_3) = P(EMG_i \mid \Theta_1) \times P(EMG_i \mid \Theta_2) \times \ldots \times P(EMG_i \mid \dot{\Theta}_3). \quad (4)$$

The assumption of independence for angles and angular velocities across the joints of a finger is not altogether valid. To account for relationships among the kinematic parameters, however, would have required a substantially more complex process for the computations. Consequently, we opted for a more tractable form for predicting EMG represented by Equation 4 at the possible expense of some loss in accuracy and theoretical rigor.

Equation 4 was applied separately to determine the muscle activity pattern for each of the three muscles (i.e. for EMG_1, EMG_2, and EMG_3). Once the probabilistic relations between joint kinematics and muscle activity had been established by application of Bayes' theorem on the training data, a set of new joint angles and angular velocities could be entered into the algorithm in order to predict the associated patterns of muscle activity (see Figure 1). For convenience, in the present case, the new set of kinematic data were obtained from the same subject from whom the training data were obtained. In theory, however, any set of desired joint trajectories could be used to predict muscle activity patterns. Ten second segments of kinematic data recorded during a variety of finger movements (but not used in the training of the algorithm) were used as inputs to the Bayes' algorithm for prediction of muscle activity. Longer segments were

not used because of limitations in memory associated with generation of timing files needed to control the muscle stimulators.

Muscle Stimulation

The predicted patterns of muscle activity that were based on the desired movement trajectories were converted into frequency-modulated trains of constant current pulses. In order to reduce computation time, successive 100 ms epochs of predicted muscle activity were consolidated into a single average value. Stimulus frequency was then linearly related to the amplitude of the average muscle activity and was held constant over the 100 ms period. Stimulus frequencies ranged from 10 - 50 Hz for predicted EMG values from 20% - 100% of the peak EMG obtained in the training set. Stimulus frequencies between 10 and 50 Hz roughly correspond to the range of firing rates recorded in human motor units during voluntary contraction (Bellemare et al. 1983). The long time constant associated with the low-pass filtering of the rectified EMG led to a relatively slow decay of the EMG following a burst such that the filtered EMG often did not reach baseline levels before the onset of a subsequent burst. Therefore, in order to avoid continuous stimulation of muscle due to this filter-induced prolongation of EMG, activity levels below an arbitrarily chosen threshold value of 20% of the peak EMG were not converted into a stimulus frequency.

In separate sessions on 5 subjects (one of whom, subject A, was the subject from whom the training data were obtained), tungsten microelectrodes with ~ 3 mm of insulation removed from the tip were placed into the same muscles recorded from during the training session. Joint angle transducers were applied in the same way as described above. The electrodes were connected to three independent stimulators and isolation

units. The amplitude of the current pulses (1 ms in duration) were then adjusted independently for each stimulator. These adjustments were made while delivering 1 s trains at 30 Hz to each muscle. Once a stimulus intensity was found that evoked, based on subjective criteria, a moderately brisk movement that spanned about 50 - 75% of the joint range of motion, the stimulus intensity was then maintained at that level for the remainder of the experiment.

Three channels of a digital-analog converter (Spike2, Cambridge Electronics Design UK) were then used to deliver pulse sequences associated with the desired movements to trigger the three stimulators. During these trials, subjects were encouraged to relax the hand and not to resist the movements generated by the stimulators. The subjects were not informed of the specific type of finger movements that were to be evoked. The initial resting configuration of the finger was not specified. Five trials were evoked for each of 5 types of movements (tapping, pulling, pushing, tapping followed by pulling, and pushing followed by tapping). Each trial consisted of a 10 s set of three pulse sequences delivered simultaneously to the three muscles.

Data Analysis

The resulting evoked movements were recorded using three position transducers as previously described. The joint angle trajectories of both the evoked and desired movements were normalized such that the maximum joint angle within each trial was set to 100% and the minimum joint angle was assigned a value of 0%. The evoked joint angles were then compared to the desired joint angles by calculating the root mean square (RMS) difference over the 10 s trial. The average RMS error for each subject was calculated over 5 trials for each joint and movement. Statistical analysis of RMS error

was performed using a two-way repeated measures ANOVA with joint and desired-movement type as factors. Post-hoc assessment of significant differences across levels within a factor was performed using a Tukey test. Differences among means were considered to be significant for $P < 0.05$.

RESULTS

Figure 3 shows a segment of the training data consisting of the unprocessed EMG signals and the corresponding rectified and smoothed EMG (RS-EMG) signals from the three muscles, the three joint angle trajectories, and the three joint angular velocities obtained while the subject (subject A) performed unrestrained movements of the middle finger. In this example, as was often the case, the kinematic pattern for the PIP and DIP joints were very similar.

Insert FIGURE 3 about here

The method by which conditional probability distributions were constructed from data like those shown in Figure 3 is depicted as a schematic diagram in Figure 4. For example, all the joint angle values (arrows, Fig. 4A) for joint 1, θ_1, associated with an activity level of 20% of the peak EMG in muscle 1 (horizontal line, Fig 4A) were used to generate the conditional probability distribution, $P(\theta_1|EMG_1 = 20\%)$, shown in Figure 4B. To aid in visualization, this distribution was then represented as a band of colored elements with hot colors indicating high probability and cool colors representing low probability. The resulting distribution was then plotted as one vertical band on the joint probability distribution plane, $P(\theta, EMG)$, shown in Figure 4C. This process was repeated for each 1% increment in EMG amplitude to fill the entire space defined by the joint probability distribution in Figure 4C. Once completed, the color at any location on

this two-dimensional plot indicated the probability that muscle 1 attained a particular value of EMG when joint 1 was at the specified angle.

Insert FIGURE 4 about here

A set of six joint-probability distributions were generated for each of the three EMG signals. From these joint probability distributions, it was possible to predict the pattern of EMG activity given a new set of desired movements. The process by which this was done is illustrated for one instant in time in Figure 5. For clarity, only two of the six joint probability distributions for one muscle, namely, MCP angle versus ED3-EMG, and MCP velocity versus ED3-EMG, are shown as the colored panels in Figures 5A and 5B, respectively. The leftmost panel in Figure 5A depicts a section of the desired angular trajectories for the MCP joint. The corresponding desired angular velocity of the MCP joint is shown in the leftmost panel in Figure 5B. In the present study, desired movements were obtained in a separate set of trials in which a subject was instructed to perform different types of finger movements. The vertical line on the desired movement trajectories represents the time at which a prediction of the EMG was to be made. The long horizontal arrows indicate the specific values of the desired MCP angle and MCP angular velocity at that instant.

Insert FIGURE 5 about here

The conditional probabilities associated with the specific values of the desired kinematics at the time instant in question (i.e., the regions of the color plots within the thin rectangles) are redrawn as a histograms immediately to the right of the color plots in Figure 5. These histograms represent the conditional probability that a kinematic parameter, θ, will attain a specified value, y, given different levels of EMG, namely, $P(\theta$

= y | EMG). Then, in accordance with Bayes' theorem, these histograms were multiplied by the overall probability of encountering different levels of EMG during the training trial, i.e., P(EMG) (Figure 5C). In this case, the probability distribution, P(EMG), was relatively uniform over different values of EMG. Consequently, the shapes of the resulting distributions, $P(\theta = y|EMG) \times P(EMG)$, were similar to the original $P(\theta = y|EMG)$ distributions. For normalization purposes, these resultant probability distributions were then divided by the total probability in that distribution, Σ (see equation 3). The normalized distributions, $P(\theta = y|EMG) \times P(EMG) / \Sigma$ are shown in the rightmost panels of Figures 5A and 5B. The total probability across each of these normalized distributions has a value of 1.0. This ensured that each kinematic parameter provided equal weight in the prediction of EMG.

Bayes' theorem specifies that the normalized distributions in the rightmost column of Figure 5 are equivalent to the conditional probability of observing various levels of EMG given the specified value of the kinematic parameter, namely, $P(EMG | \theta = y)$ (equation 3). Therefore, to estimate the most likely value of the EMG given the simultaneous occurrence of the specific values of MCP angle and MCP angular velocity, the normalized histograms in the rightmost column were multiplied together (equation 4). The outcome of that multiplication is shown in Figure 5D. A measure of the central tendency of that distribution was calculated (mean, arrow head) and that value was then used as the predicted value of the EMG for that time instant. In the present application of Bayes' theorem, six kinematic parameters (angle and angular velocity for each of three joints) were actually used to predict the most likely value of the EMG. This process was then repeated at each increment in time over the duration of the desired movement trial.

The same procedure was carried out separately to predict the EMG activity for each of the three muscles.

The landscape of the joint probability distribution for MCP joint angle versus EMG was relatively uniform over most of the space defined by the colored plane in Figure 5A. This was also the case for the PIP and DIP joints. Consequently, the shapes of the conditional probability distributions associated with different values of joint angle were similar. Hence, the ability of joint angle, by itself, to predict different levels of EMG was poor. This was in contrast to joint angular velocity in which a systematic change in the conditional probability distribution occurred for different values of angular velocity. For example, in Figure 5B, as extension angular velocity increased (i.e. increasing negative values), there was a progressive shift in the probability density toward higher values of ED3 EMG. Therefore, the inclusion of joint angular velocity was important for prediction of EMG.

Once EMG activity for a desired set of kinematic data was predicted, the EMG signal was converted into a frequency-modulated pulse pattern. Figure 6A shows a typical predicted EMG signal derived from the process outlined in Figure 5 for a push-tap movement. For comparison, Figure 6A also shows the actual ED3 EMG signal recorded (but not used in the prediction) during the trial to obtain the desired movements. Although not systematically analyzed, the correspondence between actual and predicted EMG was usually quite good with RMS errors normally less than 10%. The dashed horizontal line indicates the threshold level for converting predicted EMG into stimulus pulses. Figure 6B shows the stimulus pulse pattern resulting from the predicted EMG signal shown in Figure 6A. Stimulus frequency was a linear function of the EMG

amplitude such that the greater the amplitude of the EMG, the higher the frequency of pulses.

Insert FIGURE 6 about here

The stimulation pulse patterns derived from the predicted EMG signals for the three muscles were used to trigger three separate stimulators. In general, the resulting evoked movements were highly consistent over repeated trials. For example, Figure 7A shows the angular displacement of the MCP joint superimposed for five repeated trials of evoked tapping movements in one subject (subject C). The reproducibility of the movement was good indicating that factors such as fatigue or electrode movement did not noticeably affect the evoked responses over the course of the experiment. Indeed, in only one case in a different subject (subject A) did the pattern of evoked movement change markedly over the course of five trials for one type of movement. This was probably due to migration of one of the electrodes outside the target muscle. In this case, only the first two trials were used in the analysis. For all other movement types and subjects, all five trials were included in the analysis.

Figure 7B compares the joint angle trajectory for one of the trials of evoked movement shown in Figure 7A with the desired trajectory. Before normalization, a relatively constant bias in joint angle between desired and evoked movements led to a relatively large RMS error (38% of the maximum angular displacement of the desired movement) despite similarities in the underlying pattern of motion. After normalization (Figure 7C), the correspondence between desired and evoked movements was good as reflected in the comparatively low RMS error value of 13%.

Insert FIGURE 7 about here

Figure 8 shows an example of normalized evoked and desired trajectories for the three joints of the finger in one subject (subject B) for a movement that involved a transition from tapping into pulling at about 4.5 s into the trial. During the initial tapping portion of the trial, all three joints moved more or less in phase – flexing and extending together. Little angular displacement, however, was evoked at the DIP joint during this phase because the predicted level of EMG activity for the FDP (the only muscle that acts to flex the DIP joint) was less than the threshold level (20% of the peak EMG) set for conversion into stimulus pulses. At the time of transition from tapping to pulling (~4.7 s), a brisk extension of the MCP joint altered the phase relation among the joints such that extension of the MCP joint then occurred while the PIP and MCP joints were flexing. This subtle transition in the phase relation among the joints was reproduced with good fidelity in the evoked movement. During the latter pulling phase of the movement, the greatest discrepancy between evoked and desired movement was in the PIP joint where the depth of flexion was shallower for the evoked compared to the desired movements. A good match, however, between evoked and desired movements occurred during this phase for the MCP and DIP joints . Over the trial, RMS errors were 15.4%, 19.1%, and 15.4% for the MCP, PIP, and DIP joints, respectively.

Insert FIGURE 8 about here

An issue of particular interest in the present study was whether prediction of muscle activity based on kinematic and EMG measurements taken from one subject could be used to generate desired movements in other subjects. In most cases, the pattern of evoked movements were similar across subjects. For example, Figure 9 shows evoked trajectories of the MCP joint for the five subjects during trials involving tapping motion.

Also, superimposed on these traces is the desired trajectory. Qualitatively, there was a good correspondence in the pattern of evoked movements across the five subjects particularly in the timing of transitions from flexion to extension. Some minor difference existed, however, in the relative magnitudes of the movements across subjects at different phases of the trial. Quantitatively, the match between the desired and evoked movements in these trials was quite good for all subjects with RMS errors ranging from 12% - 17%. Furthermore, the correspondence between evoked and desired movements was no better for the subject in whom the original training data were obtained (Subject A) than for the other subjects. This was true across all movements as revealed by an one-way analysis of variance in which no statistical difference in the magnitude of RMS errors was detected across subjects for all evoked movements. Therefore, patterns of finger muscle activity predicted from data obtained in one subject can be used as templates to generate finger movements in other subjects that are reasonably close to desired movements. Whether this holds for more complex movements involving more muscles and joints is yet to be determined.

Insert FIGURE 9 about here

In order to evaluate the overall performance of the Bayes' stimulation technique, the RMS errors between the desired and evoked movements for all five subjects are summarized in Figure 10. For each subject , the average RMS error over the five trials of each 10-s movement sequence was calculated for each joint. Figure 10A shows the mean RMS error and standard deviation for the five types of movements tested (tap, push, pull, push-tap, tap-pull) averaged across the three joints for all subjects. The normalized RMS errors range from 17.8 – 26. 5%. Analysis of variance indicated there was a significant

difference among the mean RMS error values across the different types of movements. Post-hoc analysis revealed that the only significant difference in RMS error among movements was between tapping and pushing movements. The lower error in tapping may have been due to the relative simplicity of this movement which involved alternating flexion and extension of all three joints together whereas the pushing task required a more complex coordination with the MCP joint flexing and extending out of phase with the other two joints. Figure 10B shows the normalized error for the different joints (MCP, PIP and DIP) across all movement conditions. The average errors ranged from 21.8 – 23.8 % with no statistical difference in the amount of error measured for different joints. Overall, the errors were relatively modest suggesting that the evoked movements corresponded reasonably well to the desired movements.

Insert FIGURE 10 about here

DISCUSSION

Here we have shown that it is feasible to estimate the patterns of neuromuscular activity associated with a range of multi-joint finger movements and to use those patterns to evoke desired movements with good fidelity using electrical stimulation. The foundation of our approach was based on previous studies that have used Bayes' theorem to reconstruct features of motor behavior from the activity of neural populations (Brown et al. 1998; Zhang et al. 1998; Tresch and Kiehn, 2000). An implicit aim in those studies was to evaluate the amount of information contained within the activity of neural ensembles related to the behavior under study. In the present investigation, we inverted this approach and used Bayes' theorem to estimate the activity in a small ensemble of

muscles based on the motion of a multi-joint system. We have also shown that the probabilistic relationship between EMG and kinematics derived from one individual can be used to predict patterns of activity appropriate to control muscles in other individuals. The practical importance of this finding is that a functional electrical stimulation system using such a probabilistic control strategy could be trained on an able-bodied subject and then be deployed in paralyzed individuals.

The relationship between muscle activity and joint kinematics has been explored previously using other analytical techniques. One approach has been to predict muscle force from EMG activity using Hill-type models of muscle dynamics (Hof and Van den Berg, 1981; Olney and Winter, 1985; Winters and Stark, 1987; Soechting and Flanders, 1997). Predicted muscle forces are then used as inputs to a linked-segment model of a joint system to estimate joint kinematics using classical equations of motion (Zajac and Gordon, 1989; Kashima et al. 2000). While this type of approach has provided an important framework for understanding the control of limb movements, such analytical methods are extremely complex, even for one- or two-joint systems (Winters and Stark, 1987; Zajac and Gordon, 1989), and are susceptible to several sources of error (Soechting and Flanders, 1997).

Recently, artificial neural networks have been implemented in an attempt to predict limb trajectory from muscle activity (Cheron et al. 1996, Au and Kirsch 2000). From a practical standpoint, the advantage of this approach is that there is no need to specify an explicit algorithm which represents the complex set of interactions by which activity in several muscles is transformed into movement of a limb possessing multiple degrees of freedom. Instead, the interconnected elements that comprise the artificial

neural network learn relationships among a set of input and output variables when trained with example data. Such neural networks have been shown to yield excellent predictions of complex arm movements based on EMG activity recorded from several muscles (Cheron et al. ,1996; Au and Kirsch, 2000).

We followed a similar approach to that involving artificial neural networks in that no attempt was made to represent the internal mechanisms that underlie the relationship between muscle activity and movement. However, unlike the studies of Cheron et al . (1996) and Au and Kirsch (2000), our goal was to predict muscle activity patterns from movements rather than to predict movement from muscle activity. For this purpose, we used Bayes' theorem to ascertain the most likely value of muscle activity given a set of kinematic variables recorded from multiple joints of a finger. Furthermore, we extended the work of Cheron et al. (1996) and Au and Kirsch (2000) in that we used predicted muscle activity associated with desired movements to drive muscle stimulators in order to artificially elicit finger movements. The evoked movements were reasonably similar to the desired movements (mean RMS error ranged from 18- 26%) suggesting that this approach ultimately might serve as a useful strategy in attempts to restore movement in paralyzed individuals.

Disparity between desired and evoked trajectories in the present study arose primarily due to two categories of error: 1) those associated with prediction of muscle activity from joint kinematics, and 2) those related to transformation of predicted muscle activity into actual muscle activity through electrical stimulation. Overall, errors associated with prediction of EMG patterns from joint kinematics were relatively minor (Figure 5A). One probable cause for errors associated with this first category was that we

did not obtain EMG recordings from some of the intrinsic muscles of the hand (such as 2nd and 3rd dorsal interossei) which can assist in flexing the MCP joint of the middle finger. Their role was likely to be particularly important during movements in which the MCP joint was flexed while the PIP and DIP joints were extended, such as occurs during different phases of pushing and pulling tasks. These movements usually were those associated with the largest RMS errors (Figure 10A). Therefore, in the training data, there were some movements that could not be readily accounted for in the activity of the muscles from which we did record. Inclusion of additional muscles in the training set should help to further reduce errors in the kinematic-based prediction of muscle activity patterns.

The major source of error in the present study, however, was likely related to the attempt to artificially recreate the active state of the muscle developed during voluntary contraction using a frequency-modulated pulse train delivered to the muscle through a single electrode. A number of simplifying assumptions and approximations were required in order to implement such a transfer function. In natural muscle contraction, the force exerted by a muscle is dependent upon the number of muscle fibers recruited and on the rate of action potentials imposed by the motor neurons on the active fibers (Fuglevand et al 1993). The muscle fibers are organized into motor units which are groups of spatially dispersed fibers innervated by branches of the same motor axon. Variation in the strength of contraction is brought about by concurrent change in both recruitment and rate coding of motor units. The intensity of the electromyogram detected with large surface-area electrodes is also influenced by both recruitment and rate coding such that a fixed (and practically linear) relation exists between muscle force and EMG

(Fuglevand et al. 1993). Accordingly, the magnitude of the EMG provides an index of the active state of muscle which in turn is related to its mechanical (force) output.

The conversion of a predicted level of EMG into muscle activation in the present study involved delivery of current pulses through intramuscular electrodes. Because only one electrode was placed in each muscle, and because the magnitude of the stimulus current was held at a fixed level, variations in muscle activity were brought about entirely through changes in rate coding. While this method was relatively simple to implement, it did not emulate the actual process by which muscle activity is modulated. The inclusion of a means to concurrently vary recruitment and rate coding, for example, by altering both the amplitude and frequency of the delivered current pulses, would likely improve the reproduction of the active state of the muscle, and thereby, enhance the match between desired and evoked movements.

Nevertheless, the overall performance of the present approach was satisfactory in reproducing desired movements and would seem to justify further exploration and improvement of the Bayes' stimulation method. One future direction would be to include contact-force signals, perhaps mediated through tactile sensors, that together with kinematic signals could be used to predict EMG activity associated with tasks that involve interaction with external objects. Another logical extension of the current method would be to expand the number of muscles included in the algorithm in order to predict muscle activity associated with a wide range of movements of an entire limb. A major obstacle, however, to the practical implementation of such a system relates to how a paralyzed individual would supply the desired movement trajectory as input to the trained Bayes' algorithm. One possible solution would be to provide a menu of stored

desired movements from which the patient could select using non-paralyzed muscles. This approach, while feasible, would not take advantage of the flexible nature of the Bayes' method.

A promising alternative would be to decipher the desired movement trajectory directly from ensembles of neurons in the cerebral cortex (Nicolelis, 2001). Previous work in non-human primates has indicated that the activity of populations of neurons in the primary motor, pre-motor, and parietal cortices can be used to predict the intended direction of hand motion during reaching movements toward targets distributed in extrapersonal space (Georgopolus et al. 1986, 1988; Kalaska et al. 1990; Schwartz, 1993; Kakei et al. 1999, 2001, Wessberg et al. 2001). Moreover, Chapin et al. (1999) and Wessberg et al. (2001) have shown that it is possible to interpret the cortical code for a desired or actual movement and to use that brain-derived signal to control a robotic device in real-time. Therefore, desired trajectories extracted from cortical recordings conceivably could be used as inputs to the Bayes' stimulation method to produce movements in an arm and hand instead of in a robot (Hoffer et al. 1996). Such an integrated system would restore movement and independence to paralyzed individuals.

References

Au ATC, Kirsch RF (2000) EMG-based prediction of shoulder and elbow kinematics in able-bodied and spinal cord injured individuals. IEEE Trans Rehab Eng 8:471-480.

Bellemare F, Woods JJ, Johansson R, Bigland-Ritchie B (1983) Motor-unit discharge rates in maximal voluntary contractons of three human muscles. J Neurophysiol 50:1380-1392.

Brown EN, Frank LN, Tang D, Quirk MC, Wilson MA (1998) A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. J Neurosci 18: 7411-7425.

Chapin JK, Moxon KA, Markowitz RS, Nicolelis MAL (1999) Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. Nature Neurosci 2:664-670.

Cheron G, Draye J-P, Bourgeios M, Gaëtan L (1996) A dynamic neural network identification of electromyography and arm trajectory relationship during complex movements. IEEE Trans Biomed Eng 43:552-558.

Deadwyler SA, Hampson RE (1997) The significance of neural ensemble codes during behavior and cognition. Ann Rev Neurosci 20:217-244.

Fuglevand AJ, Winter DA, Patla AE (1993) Models of recruitment and rate-coding organization in motor-unit pools. J Neurophysiol 70:2470-2488.

Georgopoulos AP, Schwartz AB, Kettner RE (1986) Neuronal population coding of movement direction. Science 233:1416-1419.

Georgopoulos AP, Schwartz AB, Kettner RE (1988) Primate motor cortex and free arm movements to visual targets in three-dimensional space. II. Coding of the direction of movement by a neuronal population. J Neurosci 8:2928-2937.

Häger-Ross C, Schieber MH (2000) Quantifying the independence of human finger movements: comparison of digits, hands, and movement frequencies. J Neurosci 20:8542-8550.

Hepp-Reymond M-C, Huesler EJ, Maier MA (1996) Precision grip in humans: temporal and spatial synergies. In: Hand and Brain (Wing AM, Haggard P, Flanagan JR, eds.), pp 37-68. San Diego: Academic Press.

Hof AL, Van den Berg J (1981) EMG to force processing. I: An electrical analogue of the Hill muscle model. J Biomech 14:747-758.

Hoffer JA, Stein RB, Haugland MK, Sinkjær T, Durfee WK, Schwartz AB, Loeb GE, Kantor C (1996) Neural signals for command control and feedback in functional neuromuscular stimulation: a review. J Rehab Res Develop 33:145-157.

Hoshimiya N, Naito A, Yajima M, Handa Y (1989) A multichannel FES system for the restoration of motor functions in high spinal cord injury patients: a respiration-controlled system for multijoint upper extremity. IEEE Trans Biomed Eng 36:754-760.

Kakei S, Hoffman DS, Strick PL (1999) Muscle and movement representations in the primary motor cortex. Science 285:2136-2139.

Kakei S, Hoffman DS, Strick PL (2001) Direction of action is represented in the ventral premotor cortex. Nat Neurosci 4:1020-1025.

Kalaska JF, Cohen DAD, Prud'homme M, Hyde ML (1990) Parietal area 5 neuronal activity encodes movement kinematics, not movement dynamics. Exp Brain Res 80:351-364.

Kashima T, Isurugi Y, Shima M (2000) Analysis of a muscular control system in human movement. Biol Cybern 82:123-131.

Keith MW, Peckham PH, Thrope GB, Buckett JR, Stroh KC, Menger V (1988) Functional neuromuscular stimulation neuroprostheses for the tetraplegic hand. Clin Orthop 233:25-33.

Kilgore KL, Peckham PH, Thrope GB, Keith MW, Gallaher-Stone KA (1989) Synthesis of hand grasp using functional neuromuscular stimulation. IEEE Trans Biomed Eng 36:761-770.

Mulcahey MJ, Betz RR, Smith BT, Weiss AA, Davis SE (1997) Implanted functional electrical stimulation hand system in adolescents with spinal injuries: an evalution. Arch Phys Med Rehab 78:597-607.

Nicolelis MAL (2001) Actions from thoughts. Nature 409:403-407.

Nicolelis MAL, Ghazanfar AA, Stambaugh CR, Oliveira LMO, Laubach M, Chapin JK, Nelson RJ, Kaas JH (1998) Simultaneous encoding of tactile information by three primate cortical areas. Nature Neurosci 1:621-630.

Olney SJ, Winter DA (1985) Predictions of knee and ankle moments of force in walking from EMG and kinematic data. J Biomech 18:9-20.

Rieke F, Warland D, de Ruyter van Steveninck R, Bialek W (1997) Spikes: exploring the neural code. Cambridge, MA: MIT Press.

Robinson TL, Fuglevand AJ (1999) Independence of finger movements in normal subjects and in concert-level pianists. Soc Neurosci Abstr 25:1149.

Seifert HM, Koshland GF, Fuglevand AJ (2001) Restoration of finger movements using Bayesian reconstruction and functional electrical stimulation. Soc Neurosci Abstr 27: program 938.4.

Smith B, Tang Z, Johnson MW, Pourmehdi S, Gazdik MM, Buckett JR, Peckham PH (1998) An externally powered, multichannel, implantable stimulator-telemeter for control of paralyzed muscle. IEEE Trans Biomed Eng 45:463-475.

Soechting JF, Flanders M (1997) Evaluating an integrated musculoskeletal model of the human arm. J Biomed Eng 119:93-102.

Schwartz AB (1993) Motor cortical activity during movements: population representation during sinusoid tracing. J Neurophysiol 70 :28-36.

Tresch MC, Kiehn O (2000) Population reconstruction of the locomotor cycle from interneuron activity in the mammalian spinal cord. J Neurophysiol 83:1972-1978.

Triolo R, Nathan R, Handa Y, Keith M, Betz RR, Carroll S, Kantor C (1996) Challenges to clinical deployment of upper limb neuroprostheses. J Rehab Res Devel 33:111-122.

Wessberg, J, Stambaugh CR, Kralik JD, Beck PD, Laubach M, Chapin JK, Kim J, Biggs, SJ, Srinivasan MA, Nicolelis MAL (2000) Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. Nature 408:361-365.

Wilson MA, McNaughton BL (1993) Dynamics of the hippocampal ensemble code for space. Science 261:1055-1058.

Winters JM, Stark L (1987) Muscle models: what is gained and what is lost by varying model complexity. Biol Cybern 55:403-420.

Zajac FE, Gordon ME (1989) Determining muscle's force and action in multi-articular movement. In: Exercise and sport sciences reviews, vol. 17 (Pandolf KE ed), pp 187-230. Baltimore: Williams & Wilkins.

Zhang K, Ginzburg I, McNaughton BL, Sejnowski TJ (1998) Interpreting neuronal population activity by reconstruction: unified framework with application to hippocampal place cells. J Neurophysiol 79:1017-1044.

Figure Captions

Figure 1. Schematic diagram depicting two main stages of the experimental procedures. Stage 1) Muscle activity and kinematic signals recorded during a variety of finger movements in one subject were used determine the probabilistic relationship between muscle activity and movement using Bayes' rule. Stage 2) The relationship established by application of Bayes' rule in Stage 1 was used to predict muscle activity associated with a new set of desired kinematics. Predicted muscle activity was transformed into frequency-modulated trains of pulses which were used to control a set of muscle stimulators in order to evoke finger movements in other subjects. Evoked movements were compared to the corresponding desired movements to evaluate the accuracy of the method.

Figure 2. Experimental setup for recording joint angles and muscle activity and for stimulating muscle. The subject's arm was supported on a horizontal platform and the wrist secured in a mid-supinated position between two padded rods. Strain gauge transducers that measure joint angle were fixed over the metacarpalphalangeal, MCP, proximal interphalangeal, PIP, and distal interphalangeal, DIP, joints of the middle finger. Tungsten microelectrodes, inserted through the skin, were used to record muscle activity from or to stimulate the middle finger (digit III) compartments of the flexor digitorum profundus, FDP3; flexor digitorum superficialus, FDS3; and extensor digitorum, ED3.

Figure 3. Short segment of the data recorded in one subject (Subject A) during unrestrained movements of the middle finger that was used as input to Bayes' theorem to establish probabilistic relations between muscle activity and joint kinematics. Lower six

traces show the unprocessed electromyographic (EMG) signals and associated full-wave rectified and smoothed EMG signals (RS-EMG) recorded from digit 3 compartments of extensor digitorum (ED3), flexor digitorum superficials (FDS3), and flexor digitorum profundus (FDP3). The upper six traces show the joint angle and joint angular velocities for the MCP, PIP, and DIP joints of digit 3.

Figure 4. Schematic diagram depicting method whereby joint probability distributions were generated for each combination of kinematic parameter and electromyographic (EMG) signal. A) For each increment in magnitude of EMG activity (e.g. 20% of peak EMG, horizontal line) for muscle 1, all the corresponding joint angle (θ) values (vertical arrows) from joint 1 were used to construct the conditional probability distribution, $P(\theta_1|EMG_1 = 20\%)$ shown in B. The histogram representation shown in B was transposed into a band of colored elements based on the scale depicted on the ordinate. This colored representation is depicted in C as a vertical band for which the color of any element indicates the probability that joint 1 passed through the angle specified by the ordinate given that the EMG in muscle 1 attained a value of 20% of the peak EMG. This process was repeated for each increment in EMG magnitude in order to fill in the plane shown in C representing the joint probability distribution, $P(\theta_1, EMG_1)$.

Figure 5. Conversion of desired movements into predicted EMG by application of Bayes' theorem. For clarity, only two of the six kinematic parameters (panel A, MCP joint angle, panel B, MCP joint angular velocity) used in the prediction of EMG in one muscle (ED3) are shown . The left most traces in panels A and B depict a short time segment of desired angular trajectory and associated angular velocity for the MCP joint. Positive angular velocities represent flexion and negative angular velocities indicate

extension movements. At the time instant indicated by the vertical line, the corresponding desired values for MCP angle and angular velocity were approximately 25 deg and -200 deg/s, respectively. These values were then used to select from the joint probability distributions, P(Θ, EMG) (color plots, derived from training data recorded in one subject), the conditional probability associated with the desired joint angle and joint angular velocity (thin rectangles superimposed on the color plots). EMG values are represented as a percentage of the peak EMG value recorded during the training set and only EMG values above the threshold level for converting to stimulus pulses (20% of peak EMG) are shown. The specific conditional probabilities indicated on the color plots are redrawn as histograms, P(Θ = y | EMG), and represent the likelihood that a kinematic parameter Θ, such as MCP angle, will attain a specific value, y, e.g. 25 degrees, given different levels of EMG. These histograms were then multiplied by the overall probability of observing different levels of EMG, P(EMG) (panel C). The resultant histograms, P(Θ = y | EMG) x P(EMG), have a form similar to the original P(Θ = y | EMG) histograms because the of the relative uniformity of the P(EMG) distribution. The resultant histograms were then divided by the total probability in the histogram (Σ) to yield the normalized histograms, P(Θ = y | EMG) x P(EMG)/Σ (the rightmost plots in panels A and B). The normalized histograms were then multiplied together to yield the conditional probability distribution shown in D, $P(EMG|\ \Theta_1,\ \dot{\Theta}_1)$ which represents the likelihood of obtaining different levels of EMG given the MCP joint angle was 25 degrees and the MCP joint anglular velocity was -200 deg/s. The average value of that distribution (large arrow head) was used as the best estimate of the EMG given the specified values of the kinematic parameters. In the present application, six kinematic

parameters (angles and angular velocities for MCP, PIP, and DIP joints) were actually used to predict the most likely value of the EMG. This process was repeated for each increment in time over the entire trial.

Figure 6. An example of predicted EMG and the corresponding stimulus pulse pattern. A) Predicted EMG (thick trace) for the ED3 muscle using Bayes' Theorem based on a set of desired kinematics. Superimposed on this trace is the actual ED3 EMG (thin trace) recorded (but not used in the prediction) during the trial used to obtain desired movements. The RMS error between actual and predicted EMG was 8.8% of peak amplitude of the actual EMG. The dashed line indicates the threshold below which conversion to stimulus pulses was not carried out. B) Timing of stimulus pulses derived from the predicted EMG shown in panel A. Stimulus frequency was linearly related to the amplitude of muscle activity. Frequencies ranged from 10-50 Hz corresponding to 20% - 100% peak value of the EMG obtained in the training set.

Figure 7. Example of evoked tapping movement in one subject (Subject C). A) five superimposed traces of the evoked angular displacement of the MCP joint. B) Desired angular displacement of MCP joint (thin trace) and one trial of evoked movements of MCP (thick trace). Difference in joint angle bias and difference in magnitude of movements led to relatively large RMS error between two traces of 37.9% of maximum angular displacement of the desired movement. C) After normalization of traces in panel B, the correspondence in the pattern of motion at the MCP joint between desired and evoked trials is good (RMS error = 13%).

Figure 8. Normalized angular trajectories for MCP, PIP, and DIP joints during a trial involving a transition from tapping into pulling movement at about 4.7 s into the trial for

one subject (Subject B). Thick traces indicate evoked movements and thin traces show the desired movements. The RMS errors ranged from 15 – 19 % in this trial. The drawing of the hand was adapted from Hepp-Reymond et al. 1996.

Figure 9. Normalized angular displacement of the MCP joint during tapping motion evoked in five subjects. Each trace was obtained in a different subject. The dashed trace indicates the desired trajectory. Overall, there was a high degree of consistency in the evoked movements across subjects which all corresponded well to the desired movements. RMS errors: Subject A = 16.5%, Subject B = 11.9%, Subject C = 12.8%, Subject D = 15.4%, Subject E = 15.7%. Only data from Subject A was used to train the Bayes' algorithm and to predict the muscle activation patterns used to evoke movements in all subjects.

Figure 10. Mean and standard deviation RMS errors between desired and evoked responses across different movements and different joints A) RMS error across five different movements tapping, pushing, pulling, pushing to tapping, and tapping to pulling. Errors ranges from 18-26%. The RMS error for tapping was significantly less than that for pulling. B) RMS error across three different joints MCP, PIP, DIP. There was no significant difference in the RMS error across joints.

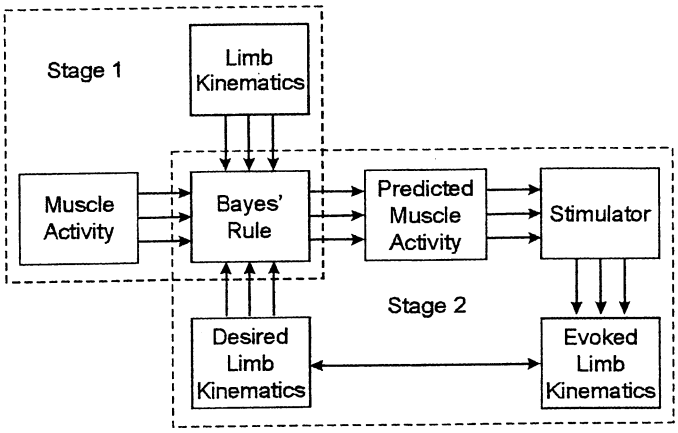


Figure 1. Schematic diagram depicting two main stages of the experimental procedures. Stage 1) Muscle activity and kinematic signals recorded during a variety of finger movements in one subject were used determine the probabilistic relationship between muscle activity and movement using Bayes' rule. Stage 2) The relationship established by application of Bayes' rule in Stage 1 was used to predict muscle activity associated with a new set of desired kinematics. Predicted muscle activity was transformed into frequency-modulated trains of pulses which were used to control a set of muscle stimulators in order to evoke finger movements in other subjects. Evoked movements were compared to the corresponding desired movements to evaluate the accuracy of the method.

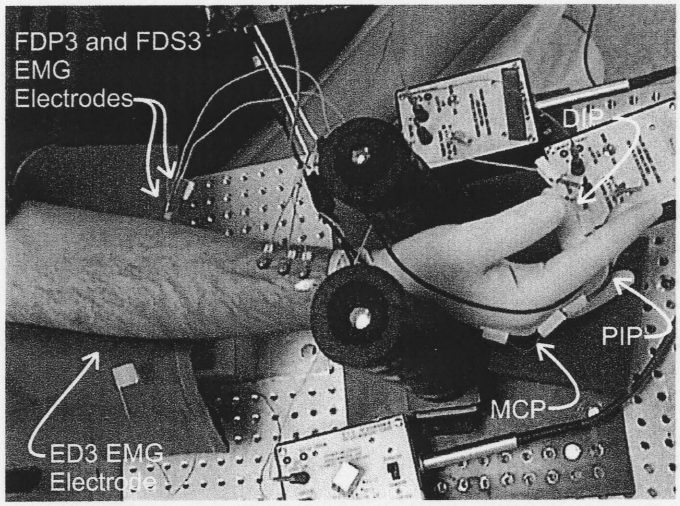


Figure 2. Experimental setup for recording joint angles and muscle activity and for stimulating muscle. The subject's arm was supported on a horizontal platform and the wrist secured in a mid-supinated position between two padded rods. Strain gauge transducers that measure joint angle were fixed over the metacarpalphalangeal, MCP, proximal interphalangeal, PIP, and distal interphalangeal, DIP, joints of the middle finger. Tungsten microelectrodes, inserted through the skin, were used to record muscle activity from or to stimulate the middle finger (digit III) compartments of the flexor digitorum profundus, FDP3; flexor digitorum superficialus, FDS3; and extensor digitorum, ED3.

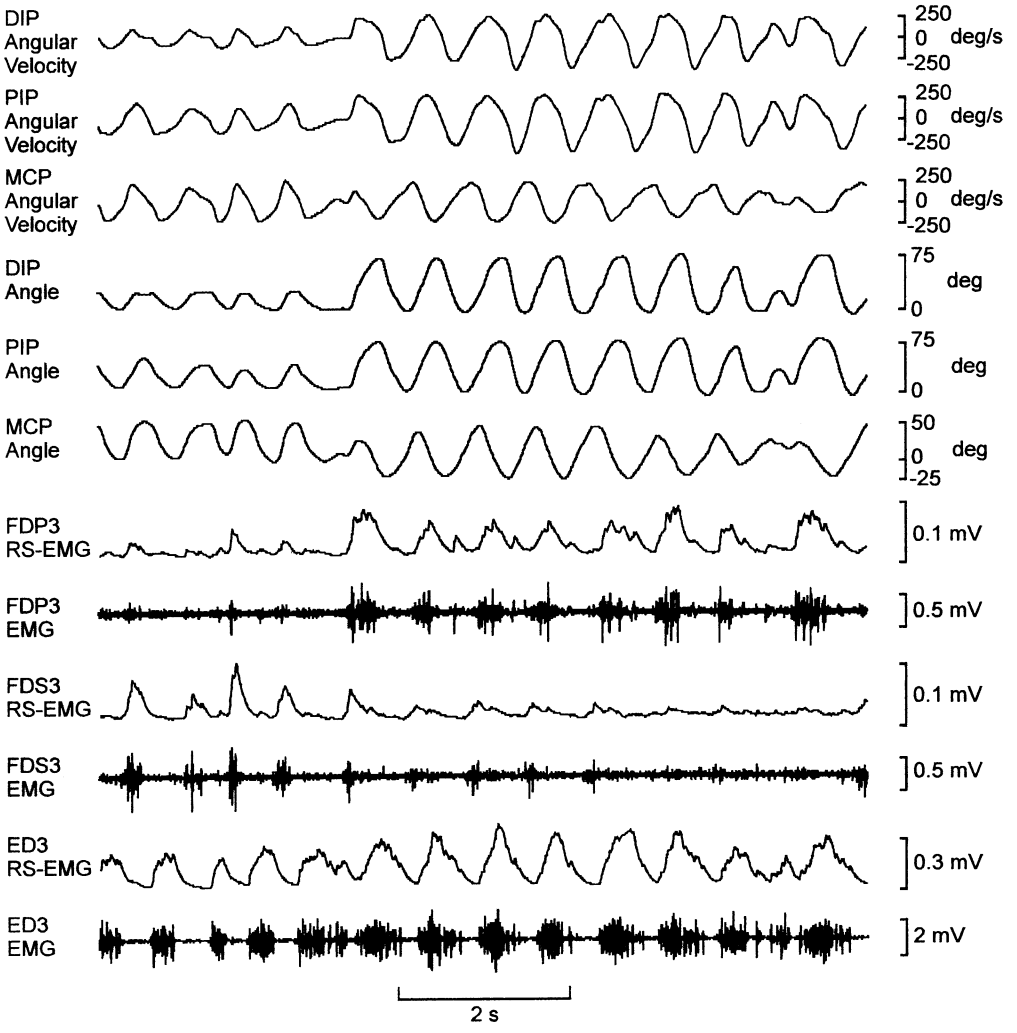


Figure 3. Short segment of the data recorded in one subject (Subject A) during unrestrained movements of the middle finger that was used as input to Bayes' theorem to establish probabilistic relations between muscle activity and joint kinematics. Lower six traces show the unprocessed electromyographic (EMG) signals and associated full-wave rectified and smoothed EMG signals (RS-EMG) recorded from digit 3 compartments of extensor digitorum (ED3), flexor digitorum superficials (FDS3), and flexor digitorum profundus (FDP3). The upper six traces show the joint angle and joint angular velocities for the MCP, PIP, and DIP joints of digit 3.

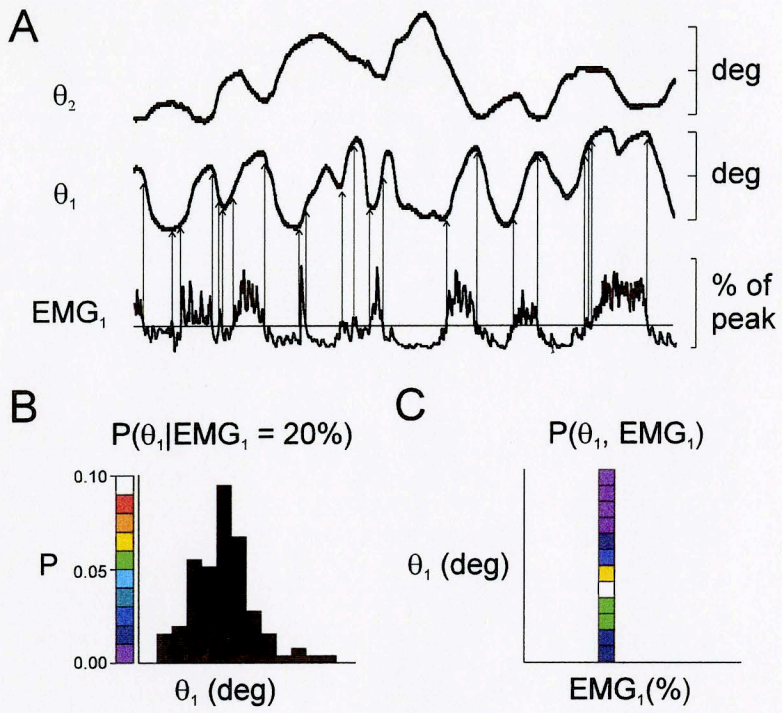


Figure 4. Schematic diagram depicting method whereby joint probability distributions were generated for each combination of kinematic parameter and electromyographic (EMG) signal. A) For each increment in magnitude of EMG activity (e.g. 20% of peak EMG, horizontal line) for muscle 1, all the corresponding joint angle (θ) values (vertical arrows) from joint 1 were used to construct the conditional probability distribution, $P(\theta_1|EMG_1 = 20\%)$ shown in B. The histogram representation shown in B was transposed into a band of colored elements based on the scale depicted on the ordinate. This colored representation is depicted in C as a vertical band for which the color of any element indicates the probability that joint 1 passed through the angle specified by the ordinate given that the EMG in muscle 1 attained a value of 20% of the peak EMG. This process was repeated for each increment in EMG magnitude in order to fill in the plane shown in C representing the joint probability distribution, $P(\theta_1, EMG_1)$.

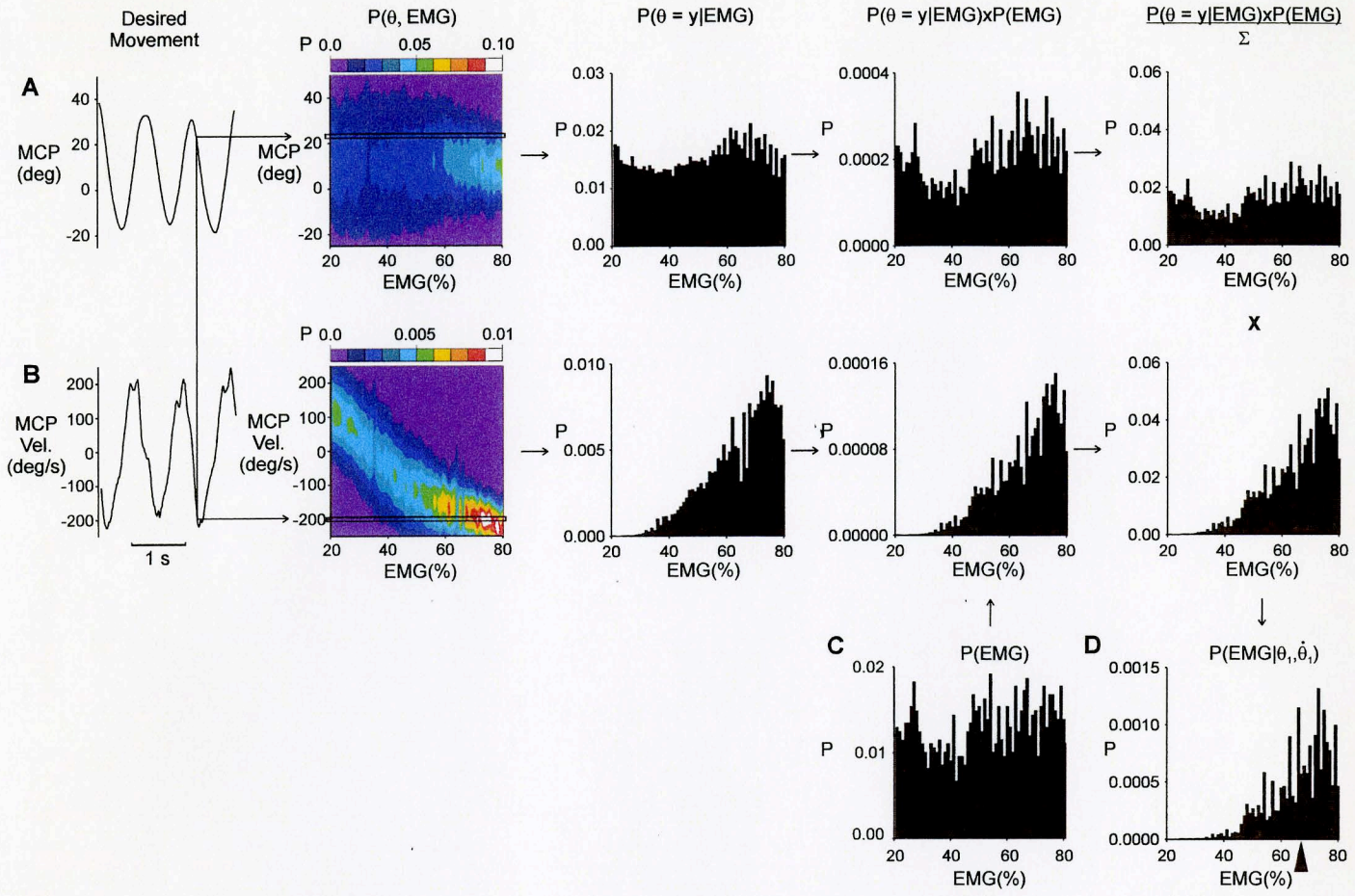


Figure 5. Conversion of desired movements into predicted EMG by application of Bayes' theorem. For clarity, only two of the six kinematic parameters (panel A, MCP joint angle, panel B, MCP joint angular velocity) used in the prediction of EMG in one muscle (ED3) are shown . The left most traces in panels A and B depict a short time segment of desired angular trajectory and associated angular velocity for the MCP joint. Positive angular velocities represent flexion and negative angular velocities indicate extension movements. At the time instant indicated by the vertical line, the corresponding desired values for MCP angle and angular velocity were approximately 25 deg and -200 deg/s, respectively. These values were then used to select from the joint probability distributions, P(Θ, EMG) (color plots, derived from training data recorded in one subject), the conditional probability associated with the desired joint angle and joint angular velocity (thin rectangles superimposed on the color plots). EMG values are represented as a percentage of the peak EMG value recorded during the training set and only EMG values above the threshold level for converting to stimulus pulses (20% of peak EMG) are shown. The specific conditional probabilities indicated on the color plots are redrawn as histograms, P(Θ = y | EMG), and represent the likelihood that a kinematic parameter Θ, such as MCP angle, will attain a specific value, y, e.g. 25 degrees, given different levels of EMG. These histograms were then multiplied by the overall probability of observing different levels of EMG, P(EMG) (panel C). The resultant histograms, P(Θ = y | EMG) x P(EMG), have a form similar to the original P(Θ = y | EMG) histograms because the of the relative uniformity of the P(EMG) distribution. The resultant histograms were then divided by the total probability in the histogram (Σ) to yield the normalized histograms, P(Θ = y | EMG) x P(EMG)/Σ (the rightmost plots in panels A and B). The normalized histograms were then multiplied together to yield the conditional probability distribution shown in D, $P(EMG|\ \Theta_1,\ \dot{\Theta}_1\ )$ which represents the likelihood of obtaining different levels of EMG given the MCP joint angle was 25 degrees and the MCP joint anglular velocity was -200 deg/s. The average value of that distribution (large arrow head) was used as the best estimate of the EMG given the specified values of the kinematic parameters. In the present application, six kinematic parameters (angles and angular velocities for MCP, PIP, and DIP joints) were actually used to predict the most likely value of the EMG. This process was repeated for each increment in time over the entire trial.

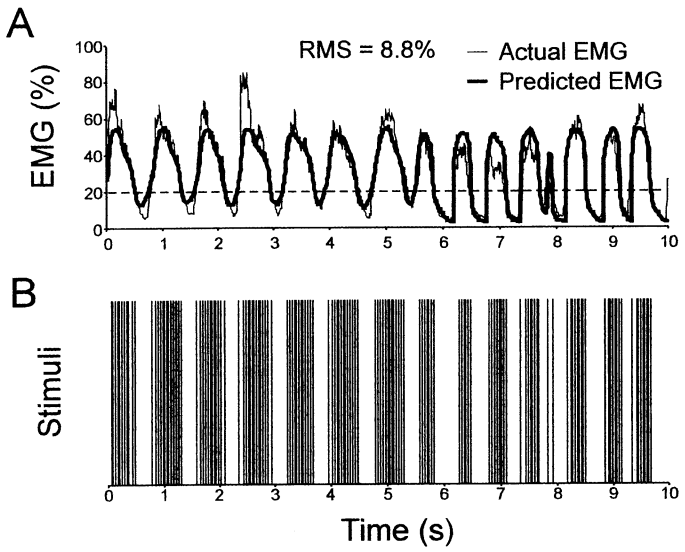


Figure 6. An example of predicted EMG and the corresponding stimulus pulse pattern. A) Predicted EMG (thick trace) for the ED3 muscle using Bayes' Theorem based on a set of desired kinematics. Superimposed on this trace is the actual ED3 EMG (thin trace) recorded (but not used in the prediction) during the trial used to obtain desired movements. The RMS error between actual and predicted EMG was 8.8% of peak amplitude of the actual EMG. The dashed line indicates the threshold below which conversion to stimulus pulses was not carried out. B) Timing of stimulus pulses derived from the predicted EMG shown in panel A. Stimulus frequency was linearly related to the amplitude of muscle activity. Frequencies ranged from 10-50 Hz corresponding to 20% - 100% peak value of the EMG obtained in the training set.

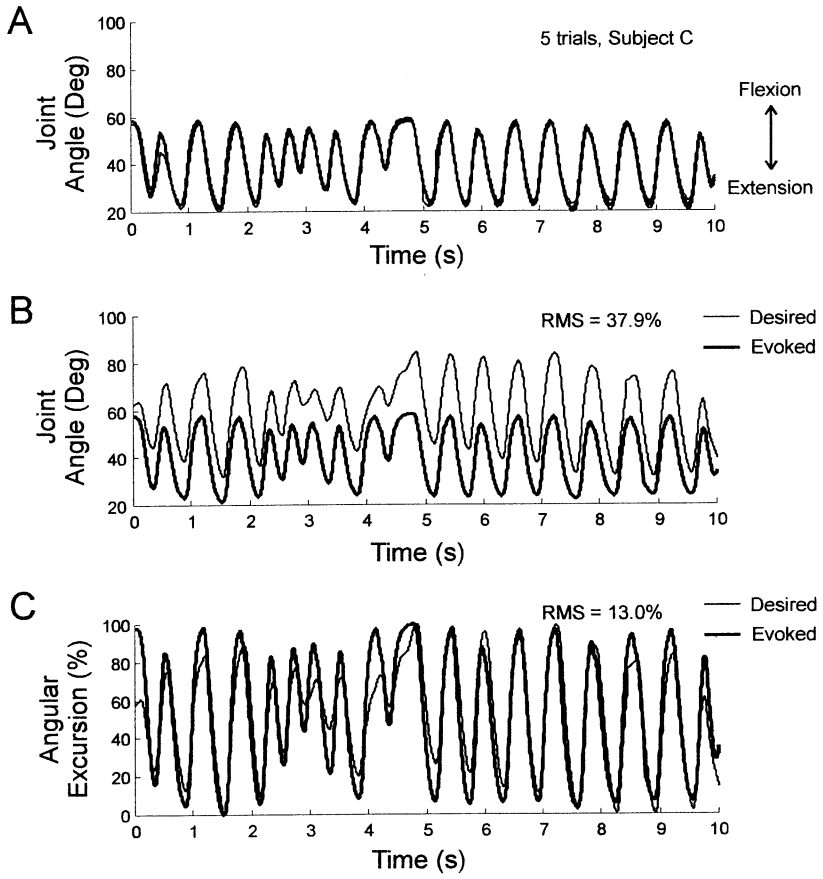


Figure 7. Example of evoked tapping movement in one subject (Subject C). A) five superimposed traces of the evoked angular displacement of the MCP joint. B) Desired angular displacement of MCP joint (thin trace) and one trial of evoked movements of MCP (thick trace). Difference in joint angle bias and difference in magnitude of movements led to relatively large RMS error between two traces of 37.9% of maximum angular displacement of the desired movement. C) After normalization of traces in panel B, the correspondence in the pattern of motion at the MCP joint between desired and evoked trials is good (RMS error = 13%).

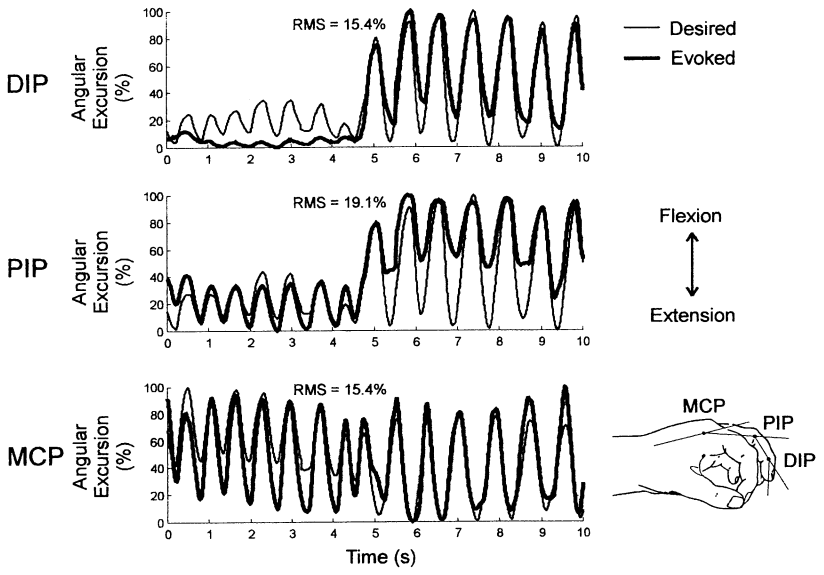


Figure 8. Normalized angular trajectories for MCP, PIP, and DIP joints during a trial involving a transition from tapping into pulling movement at about 4.5 s into the trial for one subject (Subject B). Thick traces indicate evoked movements and thin traces show the desired movements. The RMS errors ranged from 15 – 19 % in this trial. The drawing of the hand was adapted from Hepp-Reymond et al. 1996.

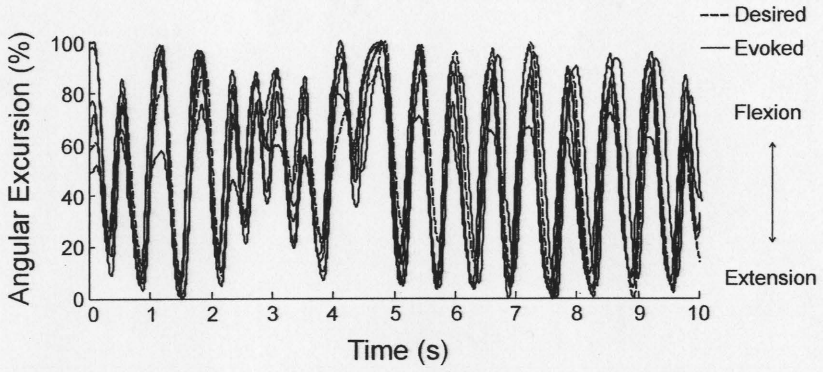


Figure 9. Normalized angular displacement of the MCP joint during tapping motion evoked in five subjects. Each trace was obtained in a different subject. The dashed trace indicates the desired trajectory. Overall, there was a high degree of consistency in the evoked movements across subjects which all corresponded well to the desired movements. RMS errors: Subject A = 16.5%, Subject B = 11.9%, Subject C = 12.8%, Subject D = 15.4%, Subject E = 15.7%. Only data from Subject A was used to train the Bayes' algorithm and to predict the muscle activation patterns used to evoke movements in all subjects.

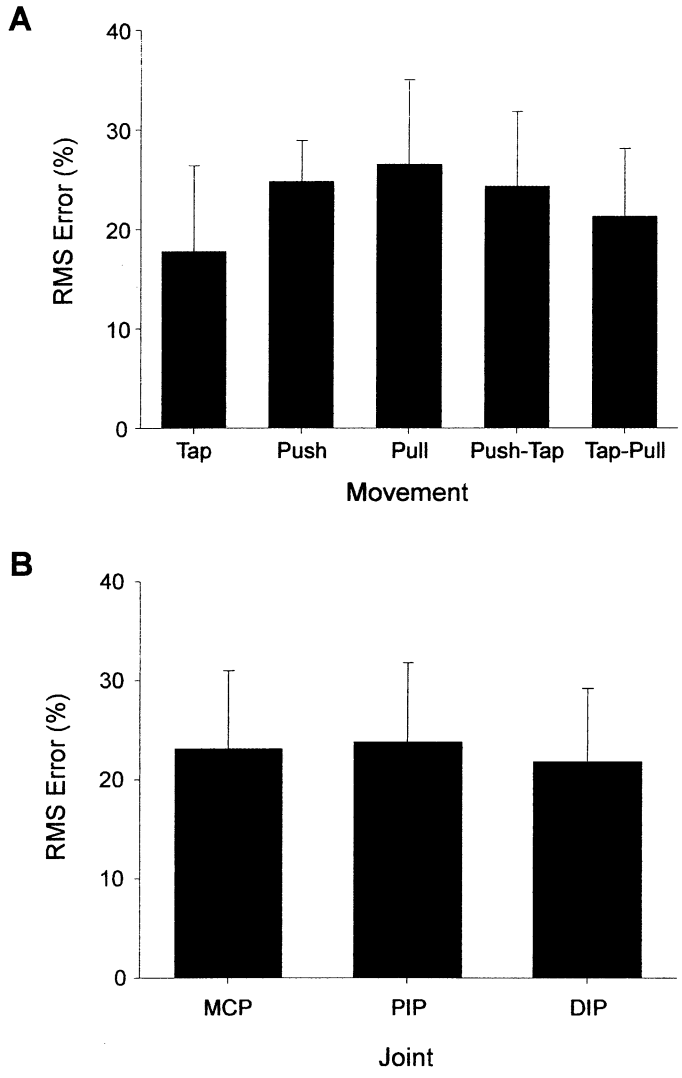


Figure 10. Mean and standard deviation RMS errors between desired and evoked responses across different movements and different joints A) RMS error across five different movements tapping, pushing, pulling, pushing to tapping, and tapping to pulling. Errors ranges from 18-26%. The RMS error for tapping was significantly less than that for pulling. B) RMS error across three different joints MCP, PIP, DIP. There was no significant difference in the RMS error across joints.

Appendix A

Summary of Experimental Protocol

Bayes' Stimulation Information and Protocol

Written by Heather Seifert on 2.16.01

Record Data

Muscle: ED3, FDP3, FDS3
    1000 gain, 30-1000Hz filter
Joints: MCP, PIP, and DIP
    1000 gain.
Configuration file: Bayesstim.s2c

Data Processing in Spike2

EMG processing: autoemg.s2s
    -INPUT: multiple EMG channels
    -OUTPUT: .txt files with relevant file names
    -Program will
        Rectify
        Low-pass filters data at 2Hz (variable)
        Subsample data at ~250Hz(variable)

Signal processing program: autopos.s2s
    -INPUT: multiple joint angle channels
    -OUTPUT: .txt files with relevant file names
    Program will:
        Low-pass filter: 2Hz(variable)
        Subsamples data at ~250Hz(variable)

*notes: you can change the filter and subsampling rate within the program. The length of the .txt files will be different for joint angle velocity. For instance, ED3 and MCP might be of length 4200 samples and the joint velocity will be 3980. The extra 220 points on the ED3 and joint angles are just zeros. This is due to the way each parameter is written to the text file with the current program. To compensate for this, the length of the emg and joint angles must be manually changed in Matlab and saved again. Annoying, but I haven't taken the time to change the Spike2 program.

Data Processing in Matlab

Program: BayesRecon.m
    -INPUT
        .txt files from autopos.s2s and autoemg.s2s
        Sampling Rate
        Beginning and ending training times
        Beginning and Ending reconstruction times
    -OUTPUT
        -predicted EMG and stimulus pulse pattern
    -Training Data
        -60 seconds long. Look for data segments without large artifact
    -Desired Data

-input about 10 seconds. Input order of files should be the same as the training sections.
for multiple start times within same file use the program Reconstruction.m

-EMG to Stim Pattern
-Ten second segments
-threshold set at 20% EMG and 19Hz frequency.
-Validation of EMG
-input actual EMG to verify the prediction with the actual EMG

*notes: There is a lot to type, so be careful not to make mistakes. This process needs to be more user friendly!! After you have developed the conditional probabilities for a particular emg, then you can use it to predict the EMG from multiple time segments with in the same position and velocity data file.

Spike2 – replaying EMG data

Load configuration file: Bayesstim.s2c
Program: emg2freq3muscles.s2s

Enter muscle files in correct order corresponding to output channels 0,1 and 2.

**notes: be very careful to input the muscle stim files in the appropriate order to stimulation the correct muscles.

Appendix B

Instructions for Data Processing and Stimulation

Data Processing Spike2

EMG PROCESSING

>Script
>Run Script
>Load and run
>Heather/spike2 programs/autoemg.s2s
choose data file
>heather/eva6/XXXX
select directory for output of files
>heather/eva6/
select directory for input (may be different that output) **this needs to be changed
>heather/eva6/
select the number of channels
>1-X
>select channel for signal processing: example: ED3
>output filename: ex: ed3.txt (be careful with file name system so a previous file won't be overwritten. Also for simplicity don't add the .txt extension)
>enter channel for output(new channel #): example:10
>enter channel new channel name: example: ed3

it will take a few seconds to process, then continue with next channel.
Then the program will say 'Signal processing complete'.
Files will be saved in the specified directory as text files.

Joint Angle Processing

>Script
>run script
>load and run
>heather/spike2 program/autopos.s2s
select file to open
>heather/eva6/
>set pathway for data output
>enter number of joint angles:
>Select channel: ex: MCP
>enter output file name: ex: mcp.txt
>enter channel for output: 10
>enter new channel name: mcp

message: new channel saved as channel XX
now differentiate each joint angle

>select channel to differentiate: (select channel that was just filtered)

>enter output name: (mcpv.txt)
>enter new channel name (i.e mcpv)
>enter new units (deg/s)

continue this for each joint.
Message at end: signal processing complete.

Vector Length Adjustment

This is an annoying process but necessary. Hopefully someone in the future can change this.

Load matlab
Check to see if you are in the correct directory.
>File
>setpath
>browse (select folder with data)

now type at command prompt

>>ls (lists your files. If there are not there check setpath again)
>>load filename
    load all the file names EMG, joint angles and velocities
>>whos
    this command will show you all of the variables that you loaded and the sizes.
    You will notice that the joint angles and EMG are longer than the velocities.
    The matlab program I wrote can not deal with this discrepancy, Therefore, you
    will need to change them to be of the same length and save them again.
>>mcp = mcp(1:X); , where X is the length of the velocity vector. Semicolon is important
>>save mcp mcp –ascii (this will save the file and yes there should be two mcp in this
    command. If you want to add an extension the the file name it should look like
    save mcp.txt mcp -ascii)

    Loop/continue this process for all joint angles and EMG.
    When this is completed, clear the workspace by typing
    >clear

Bayes Reconstruction

Load matlab
Check to see if you are in the correct directory by clicking on
File
setpath
browse (select folder with data)

now type at command prompt

>>ls (lists your files. If there are not there check setpath again)
>>BayesRecon (this is the program that predicts emg)
>enter the EMG: (this is the EMG that you will use to train the algorithm)
enter number of kinematic parameters used for prediction:
(anynumber, 3 joints + 3 velocities=6)
>enter training parameter 1: (you will be prompted for all 6 kinematic
**pay careful attention to the order in which you type the files.
You will need to type in the desired files in the same order
>enter desired data 1:
>input sampling rate: (this is the rate after subsampling. It should be the original sampling rate/10. Meaning subsampled every 10^{th} pt. The sampling rate can be found in the Spike2 Config. file for the experiment.

>enter start time of training data: (in seconds)
>enter end time of training data: (in seconds)
>enter start time of desired data: (seconds)
>enter end time of desired data: (seconds)

(now the program will work for a short while, a graph should appear with the predicted emg and stimulus pattern. The stimulation

>enter file name for output (this will save your stimulation pattern)
>enter actual EMG (this is for validation. If you don't have the original EMG then hit control C. That will stop the program.)

Continue this process for each EMG of interest. You will have to perform the above process for each new EMG.
Clear the workspace after you are done creating the stim files for multiple time segments(see below) for an EMG. To clear the workspace type: clear

Multiple Time Segments for A particular EMG

Now that you have developed the conditional probabilities for the EMG you are interested in, you can run multiple start and end times for reconstruction This minimizing the time spent typing in all the EMG and kinematic files.

In the same directory type
>reconstuction
>enter start time of desired data
>enter end time of desired data

After a few moments, a graph should appear.

>enter file name for output
>enter actual EMG

Loop for additional time segments within the same file.

Loading EMG stimulation files into Spike2 Play Waveform

This will load the files you created with Bayes into Spike2 for stimulation
Log on as administrator to get the appropriate access to Spike2. If you don't do this, you might get an error running the 1401 DAQ board.

>open spike2
>File
>load configuration
>heather/spike2 programs/Bayesstim.s3c
>script
>run script
>load and run
>heather/spike2 program/emg2freq3muscles
>set pathway for data input. (pathway for you stimulation files)
>enter filename 1. (Name must be typed in exactly otherwise the stim file won't load. Names must be typed in order according to the output stim. 1st name = port 0, 2nd name = port 1, 3rd name = port 2.
>enter length of trial. (This must be 10 for now because of system requirements. If 1401 is upgraded with more memory, the time could increase)
The configuration file will load this files. You can check that it is there by typing
>sample
>sampling configuration
>play waveform
Key = x
Label = stimulation
DACs = 0,1,2
Rate = 10000
Size = around 600kb
Cycles = 1 (you can change this to be anything)
Source = waveform script

When you load different trials, you will not notice a difference in the Play Waveform. It is suggested that you check the stimulation pattern on spike 2 before delivering it.

Run and Collect Data

Click on

>File
>new
>Data Document
The simulation process begins when the 'stim' button in the top left corner is pressed.

Appendix C

Spike2 Programs for EMG and Kinematic Data Processing

```
 This script automates the signal processing. It will ask for the EMG channels,
'rectify the channels, filter at 2HZ, and subsample at 10HZ
'Adapted from the program:
'Need to update subsampling rate. This is not asked of the user since once it is set,
'it won't be changed for a while

'Heather Seifert
'Created 2.12.01
'last update --> 2.12.01

var mc%,mc2%, x,z,Q,y1, Qrepeat,ss;  'channel query varibles
var a0, b1, si, f, i, t, y, h, s,jj,ii;           'filter variables
var ezero,hms;                                  ' bias compensator
var mvc[5],subsample%, samplingrate,rate;
var channel[5], fsave$, fsaveinput$;
var data%;
var title$;
var ch%;                                ' saving channel number


data%:=FileOpen("",0,1);
Message ("Waveform channel signal processor");
Message("Set pathway for data analysis output");
FilePathSet(fsave$);
fsave$:=FilePath$();

Message("Set pathway for data input");
FilePathSet(fsaveinput$);
fsaveinput$:=FilePath$();

subsample% := 10;

x:=0;
jj:= input("How many muscle channels",1);
subsample% := 10;

for ii:=1 to jj do
   var ret%;
   var fnam$;
   var units$;
   var chan%;

   DlgCreate("Signal Processing for EMG");
   DlgChan(1,"Select channel for signal processing",1);
   DlgString(2,"Enter output file name",9);
   DlgChan(3,"enter channel for output",128);
   DlgString(4,"Enter new channel name",9);
   ret%:=DlgShow(chan%, fnam$,ch%, title$ );

   ezero:= Count(Chan%, 0, Maxtime(Chan%));  'calculate baseline of signal

   printlog(rate);
   mc2%:=MemChan(1,0,BinSize(Chan%)*10);

   mc%:= MemChan(1,0,BinSize(Chan%));  ' creates new channel
   ChanScale (Mc%, ChanScale(CHAN%));
   ChanOffset (mc%, Chanoffset(Chan%));
   ChanUnits$ (mc%, Chanunits$(CHAN%));
   MemImport(mc%, CHAN%, 0, Maxtime());

   'makes a copy for the subsampled channel
   ChanScale (Mc2%, ChanScale(CHAN%));
   ChanOffset (mc2%, Chanoffset(Chan%));
   ChanUnits$ (mc2%, Chanunits$(CHAN%));
   MemImport(mc2%, CHAN%, 0, Maxtime());
   printlog(chanScale(chan%));
   printlog(chanoffset(Chan%));
   printlog(chanunits$(chan%));

   f:=3;

   var sub[2000];
   var work[20000];                    'array for data analysis
   var read%, read2%,stime, etime;     ' counts # of data points, start and end time of record

   stime:=0;
```

```
   etime:=maxtime();

   repeat

      read%:=ChanData(mc%, work[], stime, etime, stime); ' gets # of data points (up to 20000)
      read2%:=(read%/subsample%);
      ArrSub(work[], ezero);

      if (read%>0) then

         'rectifying
          abs(work[:read%]);

         ''digital filter
            t:= Binsize(mc%);
            a0:= (6.283 *f * t)/(1+(6.283*f*t));
            b1:= 1/ (1 + (6.283* f *t));
            For i:= 1 to (read%-1) do
               y:= (a0*work[i]) + (B1*work[i-1]);
               work[i]:=y;
            Next;


          'subsampling
         h:=-1;

         for i:=1 to len(work[])-1 step subsample% do
            h:= h+1;
            s := (work[i]);
            sub[h] := s;
         next;

         FilePathSet(fsave$); ' directory
         FileOpen (fnam$, 8, 3); ' open external text file for writing and appending
         for hms:=0 to 1999 do
         PRINT ("%6.2f\n", sub[hms]);
         next
         FileClose();

         FilePathSet(fsaveinput$); 'change directory back to data input
         MemSetItem(mc2%,0,stime,sub[:read2%]);
         MemSetItem (mc%, 0, stime, work[:read%]);
         stime:=stime + Binsize(mc%)*read%;

      endif;

   until read%<=0;

   chanshow(mc2%);
   Optimise(mc2%);

   var check, list%[32], c, Qcheck, L;


      ChanList(list%[]);

      'saving the subsampled channel mc2. Not saving the the filtered alone
      MemSave (mc2%, ch%, 1);
      ChanDelete (mc2%);
      ChanDelete(mc%);
      ChanShow(ch%);
      Optimise (ch%);

      Message("New channel saved as channel %2.0f", ch%);
      ChanTitle$(ch%, title$);


next;

Message ("Signal processing complete");

halt;
```

```
'This script automates the signal processing. It will ask for the EMG channels,
'rectify the channels, filter at 2HZ, and subsample at 10HZ
'Adapted from the program:
'Need to update subsampling rate. This is not asked of the user since once it is set,
'it won't be changed for a while

'Heather Seifert
'Created 2.12.01
'last update --> 2.12.01

var mc%,mc2%, x,z,Q,y1, Qrepeat,ss;  'channel query varibles
var a0, b1, si, f, i, t, y, h, s,jj,ii;           'filter variables
var ezero,hms;                                  ' bias compensator
var mvc[5],subsample%, samplingrate,rate;
var channel[5], fsave$, fsaveinput$;
var data%;
var title$;
var ch%;                              ' saving channel number


data%:=FileOpen("",0,1);
XRange(0,Maxtime());
Optimise(-1);
Message ("Waveform channel signal processor");
Message("Set pathway for data analysis output");
FilePathSet(fsave$);
fsave$:=FilePath$();

fsaveinput$:=FilePath$();

subsample% := 10;

x:=0;
jj:= input("How many joint angles",1);
subsample% := 10;

for ii:=1 to jj do
   var ret%;
   var fnam$;
   var chan%;

   DlgCreate("Signal Processing for Position");
   DlgChan(1,"Select channel for signal processing",1);
   DlgString(2,"Enter output file name",15);
   DlgChan(3,"enter new channel ",128);
   DlgString(4,"Enter new channel name",15);

   ret%:=DlgShow(chan%, fnam$,ch%, title$ );

   ezero:= Count(Chan%, 0, Maxtime(Chan%));  'calculate baseline of signal

   printlog(rate);
   mc2%:=MemChan(1,0,BinSize(Chan%)*10);

   mc%:= MemChan(1,0,BinSize(Chan%));  ' creates new channel
   ChanScale (Mc%, ChanScale(CHAN%));
   ChanOffset (mc%, Chanoffset(Chan%));
   ChanUnits$ (mc%, Chanunits$(CHAN%));
   MemImport(mc%, CHAN%, 0, Maxtime());

   'makes a copy for the subsampled channel
   ChanScale (Mc2%, ChanScale(CHAN%));
   ChanOffset (mc2%, Chanoffset(Chan%));
   ChanUnits$ (mc2%, Chanunits$(CHAN%));
   MemImport(mc2%, CHAN%, 0, Maxtime());
   printlog(chanScale(chan%));
   printlog(chanoffset(Chan%));
   printlog(chanunits$(chan%));

   f:=3;

   var sub[2000];
   var work[20000];                    'array for data analysis
   var read%, read2%,stime, etime;     ' counts # of data points, start and end time of record

   stime:=0;
```

```
etime:=maxtime();

repeat

   read%:=ChanData(mc%, work[], stime, etime, stime); ' gets # of data points (up to 20000)
   read2%:=(read%/subsample%);

   if (read%>0) then

      ''digital filter
         t:= Binsize(mc%);
         a0:= (6.283 *f * t)/(1+(6.283*f*t));
         b1:= 1/ (1 + (6.283* f *t));
         For i:= 1 to (read%-1) do
            y:= (a0*work[i]) + (B1*work[i-1]);
            work[i]:=y;
         Next;


       'subsampling
      h:=-1;

      for i:=1 to len(work[])-1 step subsample% do
         h:= h+1;
         s := (work[i]);
         sub[h] := s;
      next;

      FilePathSet(fsave$); ' directory
      FileOpen (fnam$, 8, 3); ' open external text file for writing and appending
      for hms:=0 to 1999 do
      PRINT ("%6.2f\n", sub[hms]);
      next
      FileClose();

      FilePathSet(fsaveinput$); 'change directory back to data input
      MemSetItem(mc2%,0,stime,sub[:read2%]);
      MemSetItem (mc%, 0, stime, work[:read%]);
      stime:=stime + Binsize(mc%)*read%;

   endif;

until read%<=0;

chanshow(mc2%);
Optimise(mc2%);

var check, list%[32], c, Qcheck, L;


   ChanList(list%[]);

   'saving the subsampled channel mc2. Not saving the the filtered alone
   MemSave (mc2%, ch%, 1);
   ChanDelete (mc2%);
   ChanDelete(mc%);
   ChanShow(ch%);
   Optimise (ch%);

   Message("New channel saved as channel %2.0f", ch%);
   ChanTitle$(ch%, title$);

   var dataChan%;
   var retur%;
   var name$;
   var units$;
   var velocity$;
   var mm%;
   DlgCreate("Differentiator");
   DlgChan(1,"Select channel to differentiate",1);
   DlgString(2,"Enter output file name",15);
   DlgString(3,"Enter units of new channel",5);
   DlgString(4,"Enter new channel name",15);
   retur%:=DlgShow(dataChan%,velocity$, units$, name$ );
   mm%:=MemChan(1,0,Binsize(dataChan%));
   ChanTitle$(mm%, name$);
```

```
      ChanUnits$(mm%, units$);
      DiffChan(dataChan%, mm%, 0,maxtime(datachan%));
      ChanList(list%[], 128);
      if list%[0]>0 then
       MemSave(mm%, list%[1]);
      ChanShow(list%[1]);
      ChanDelete(mm%);
      else
      Message("There are no more free channels.\nDelete some and try again");
      endif;

func SetArr%(arr[]);
arr[0]:=-0.2;
arr[1]:=-0.1;
arr[2]:=0;
arr[3]:=0.1;
arr[4]:=0.2;
end;

func SetScOff%(arr[], ch%)
var yHi;
var yLo;

yHi:=arr[Max(arr[])];
ylo:=arr[Min(arr[])];

ChanOffset(ch%, (yHi+yLo)/2);
ChanScale(ch%, (yHi-yLo)/10);
end;

func DiffChan(source%, dest%, locSTime, locETime)
var arrSize%;
var pts%;
var locFTime;
arrSize%:=Round((locETime-locSTime)/Binsize(source%))+1;
var arr[arrSize%];
var filtArr[5];

SetArr%(filtArr[]);

pts%:=ChanData(source%, arr[], locSTime, locETime, locFTime);
ArrFilt(arr[:pts%],filtArr[]);
ArrDiv(arr[:pts%],Binsize(source%));
SetScOff%(arr[:pts%],dest%);
MemSetItem(dest%, 0, locFTime, arr[:pts%]);

FileOpen (velocity$, 8, 3); ' open external text file for writing and appending
for hms:=0 to len(arr[])-1 do
PRINT ("%6.2f\n", arr[hms]);
next
FileClose();

return locFTime;
end;



next;

Message ("Signal processing complete");

halt;
```

Appendix D

Bayes' Reconstruction Algorithm

```
%This program uses Bayes' theorem to predict the EMG activity for a set of joint angles
%The second half of the program converts the predicted EMG into a stimulation pulse
%pattern

%Just using the first part of the program, the parameters could be reversed
%to predict the joint angle from multiple muscles.
%Also, the user can use any two parameters as long at the two parameters have
%the same sampling rate and aligned in time

%Heather Seifert
%Created: 3.22.01
%Last Update: 10.19.01

%Revision 10.19.01  I made the predicted pattern print bold (line 199)
%                   I normalized the actual EMG before calculating the RMS error (line
276)

%Revisions to be made:  Need to check out the stimulation pattern algorithm. Last time I
ran it, the stimulation pattern seems to lag the
%the predicted pattern.


%This next section will load in the data. Here are the code words.
%P( predict | train ) = P(train|predict)*P(predict)/P(train)
%Predict = data that you want predicted (Bayes)
%Train = data that will be used for training (Bayes)
%Desired = the desired data used to determine the predict data


predictfile = input('enter the EMG : ','s');
predict = (load(predictfile));
number = input('enter number of kinematic parameters used for prediction: ');
for jj=1:number
   fnamestrain{jj} = input(['enter training parameter'num2str(jj)],'s');
   train(:,jj) = round(load(fnamestrain{jj}));
end


for jj=1:number
   fnamesdesired{jj}= input(['enter desired data'num2str(jj)],'s');
   desired(:,jj) = round(load(fnamesdesired{jj}));
end

samplingrate = input('input the sampling rate');
starttime = input('input the start time of the training data');
endtime = input('input the end time of the training data');
starttime2 = input('input the start time of the desired data');
endtime2 = input('input the end time of the desired data');

startpoint = starttime*samplingrate;
endpoint = endtime*samplingrate;
startpoint2 = starttime2*samplingrate;
endpoint2 = endtime2*samplingrate;

%The following part adjusts the desired position to fit within that limits of the training
data
for ii = 1:number
jointspace(ii,1:2) = [max(train(:,ii)),min(train(:,ii))];
end

for ii = 1:number
jsdesired(ii,1:2) = [max(desired(:,ii)),min(desired(:,ii))];
end

for ii = 1:number
   if jointspace(ii,1) < jsdesired(ii,1);
      for kk = 1:length(desired)
         if desired(kk,ii) > jointspace(ii,1);
```

```
            desired(kk,ii) = jointspace(ii,1);
         end
      end
   end
end

for ii = 1:number
   if jointspace(ii,2) > jsdesired(ii,2);
      for kk = 1:length(desired)
         if desired(kk,ii) < jointspace(ii,2);
            desired(kk,ii) = jointspace(ii,2);
         end
      end
   end
end

%set the length for each of the vectors
predict =((predict(startpoint:endpoint)));
normalization = max(predict);
predict = round((predict/max(predict))*100);
%predict =round((predict(startpoint:endpoint)));
%%This LOOP STARTS BAYES

for tt = 1:number
   traindata = round(train(startpoint:endpoint,tt));
   desiredata =round(desired(startpoint2:endpoint2,tt));

   minimum = 1-round(min(train(:,tt)));
   %minimum is used for finding the minimum since the array indices can not be less than 1
   %This sets up an array to go from 1-[max(emg)+minimum]

   %%Prob(predict)
   predictval = unique(predict);
   probpredict = histc(predict, predictval)/length(predict);


   %Initializing the conditional probability array
   condprob = zeros(max(round(train(:,tt))+minimum),length(predictval));

   n=0;
   for ii=1:length(predictval)
      indexpredict = find(predict == predictval(ii));
      for jj=1:length(indexpredict)
         newtrain(jj) = traindata(indexpredict(jj));
      end
      %find the conditional prob for traindata
      for jj=1:length(indexpredict)
         if newtrain(jj) ~=.5
            indextrain = find(newtrain == newtrain(jj));
            yy = newtrain(jj);
            condprob(yy+minimum,ii) = length(indextrain)/length(indexpredict);
            trainval(yy+minimum,ii) = newtrain(jj);
            newtrain(indextrain) = .5;
          end
      end
   end %%END conditional probability loop

   %%Gaussian Smoothing
   for kk=1:length(predictval)
      S=1:size(condprob,1);
      mu = sum(condprob(:,kk).*S');
      stdev=sqrt(sum((S-mu).^2.*(condprob(:,kk)')));
      if stdev ~= 0
         for ii=1:size(condprob,1)
            cp(ii) = (1/(stdev*2.5066))*exp(-.5*([ii-mu]/stdev)^2);
         end

         %The following 4 lines will plot out the conditional probability
```

```
            %bar(condprob(:,kk)), hold on, plot(cp,'r');
            %pause
            %hold off;
            condprob(:,kk) = cp(:);

        end
    end

    %Use these lines to look at the conditional probability of each parameter
    %figure(3+tt),pcolor(condprob);
    %Shading interp;



    %%Setting up data to be used outside of loop
    condprobability{tt} = condprob;
    recondata{tt} = desiredata;
    offset{tt} = minimum;
    cp = 0;
end  %END BIG LOOP

%%TRAINING DATA NOW FINISHED, now reconstruction
SS=1;
bayes = repmat(0,1,length(predictval));

for uu=1:1:length(desiredata)
    SS = 1;
    for hh = 1:number
        SS= SS.*condprobability{hh}((recondata{hh}(uu))+offset{hh},:);
        %P(emg1, emg2, emg3|pos) = P(emg1|pos)*P(emg2|pos)*P(emg3|pos)
    end

    SS=SS.*probpredict(:)'; %This is a vector
    bayes = SS;
    S2 = sum(SS); %This is for the denominator, a scaler (normalizing)

    if S2 ~=0
        bayes = bayes/S2;
    else
        bayes(:) =0;
    end

    sums=0;

    %the following 5 lines will show you what the bayes distribution looks like
    %bar(bayes);
    %hold on
    %plot(probpredict)
    %hold off
    %pause

        %%%Finding the median value of the P(pos|emg) distribution
    for ii=1:length(predictval)
        if sums <=.5
            sums=bayes(ii)+sums;
            p=predictval(ii);
        end
    end

    %%%the mean value determines the position
    predictedvariable(uu)= p;
    bayes = repmat(0,1,length(predictval));
end

title('Predictd Variable');
time = (1/samplingrate:1/samplingrate:length(predictedvariable)*(1/samplingrate));
hold off, figure(1), subplot(2,1,1),plot(time,predictedvariable,'linewidth',2);
axis([0 10 0 100])
```

```
inputdata = predictedvariable;
begintime = 1;
triallength = 10; %input('how long do you want the trial?:    ');
datafile = input('enter file name for output: ','s');

%%Open file
fid = fopen(datafile ,'w');

if begintime == 1
   stime = 1;
else
   stime = round(begintime*samplingrate); % Time in # of points (starttime)

end

%%Normalizes the data
maximum = max(inputdata(stime:(stime+triallength*samplingrate)));
maximum = 100;
emgstimdata = round((inputdata(stime:(stime+triallength*samplingrate))./maximum)*100);

%%Creates a vector with delay times (i.e frequecies from 1-50HZ, it is a 2:1 relationship
with emg)
%%The threshold for EMG is 19%.  To change the threshold, you need to change the
loop/function.
jj = 0;
for ii=1:.5:50
   jj = jj+1;
   delay(jj) = 1/(ii*.0001);
end


%%Creates Stimulation Pulse pattern for the predicted EMG
volts=32767;
off = 0;
counter =0;

stime = 1;
while stime < triallength*samplingrate-round(samplingrate/10);

   average = sum(emgstimdata(stime:stime+round(samplingrate/10)))/length(emgstimdata
(stime:stime+round(samplingrate/10)));
   isi = delay(round(average-1));

   if isi < 1000
      fit = floor(1000/isi);
      xtratime = round(fit*isi*samplingrate/10000);
      for jj=1:fit
         fprintf(fid, '%d\n', volts);
         for pp=1:isi
            fprintf(fid, '%d\n', off);
         end
      end
   else
      for jj=1:isi
         fprintf(fid, '%d\n', off);
      end
      xtratime= round(samplingrate/10);
   end
   xtratime;
   stime = stime+xtratime;
   counter = counter+1;

end
stimdata = load(datafile);
time = (.0001:.0001:.0001*length(stimdata));
figure(1), subplot(2,1,2), plot(time, stimdata), axis([0 10 0  1])
fclose(fid);
```

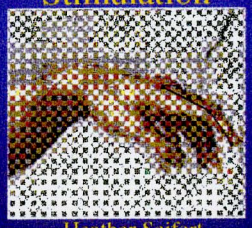
```
text(0,-.2,['trainemg=' predictfile])
text(3, -.2, ['strain' num2str(starttime)])
text(5, -.2, ['etrain' num2str(endtime)])
text(7, -.2, ['sdesire' num2str(starttime2)])
text(9, -.2, ['edesire' num2str(endtime2)])

%Plots real EMG on top of predicted.
emg = input('enter actual EMG: ','s');
emgdata = load(emg);
emgdata = round(emgdata(startpoint2:endpoint2)*100/normalization);
time = (1/samplingrate:1/samplingrate:length(predictedvariable)*(1/samplingrate));
k= (predictedvariable'-round(emgdata));
rmserror = sqrt(sum(k.*k)/(length(predictedvariable)-1));
figure(1), subplot(2,1,1), hold on, plot(time, emgdata,'r');
hold off
axis([0 10 0 100])
text(0,-20,date)
text(4, -20, ['emg=' emg])
text(8, -20, ['rmserror=' num2str(rmserror)])
```

Appendix E

Master's Presentation

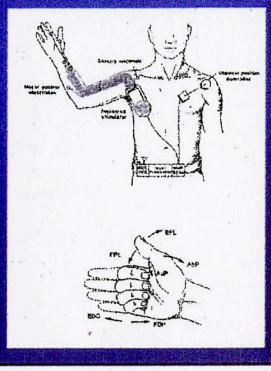
Reanimation of Finger Movement using Functional Electrical Stimulation



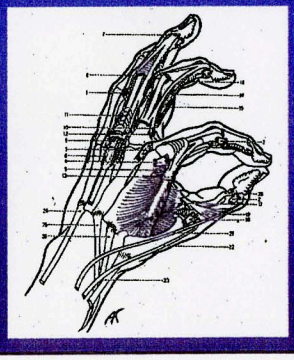
Heather Seifert

Biomedical Engineering
Interdisciplinary Program

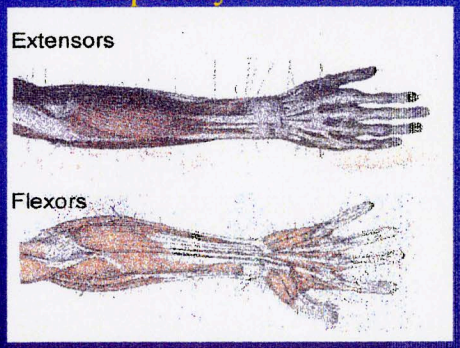
Functional Electrical Stimulation



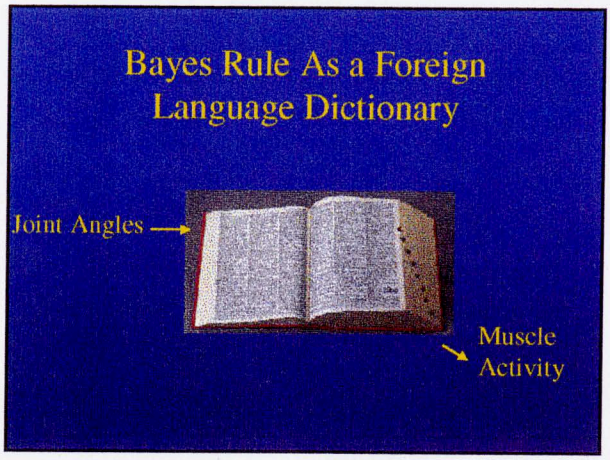
Complexity of Hand



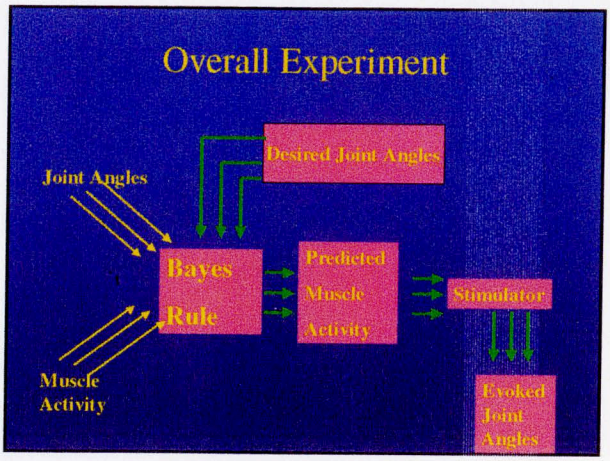
Complexity of Muscles



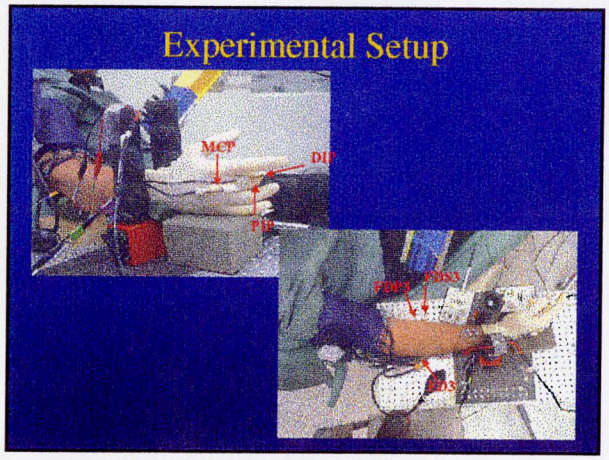
Bayes Rule As a Foreign Language Dictionary



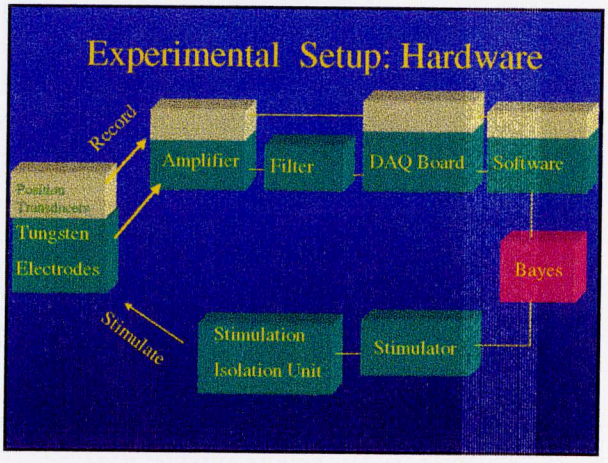
Overall Experiment



Experimental Setup



Experimental Setup: Hardware



Bayes Rule

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

$$P(B) = \sum_{B} P(B|A)P(A)$$

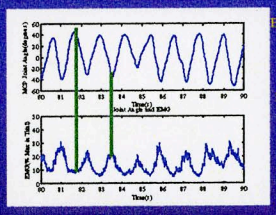
Multiple Parameters

$$P(A|B_1) = \left(\frac{P(B_1|A)P(A)}{\Sigma P(B_1|A)P(A)}\right) * \left(\frac{P(B_2|A)P(A)}{\Sigma P(B_2|A)P(A)}\right) * \left(\frac{P(B_n|A)P(A)}{\Sigma P(B_n|A)P(A)}\right)$$

Applied to Muscle Activity and Joint Angles

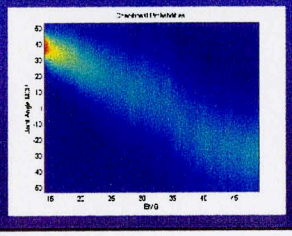
$$P(EMG|\theta_n) = \frac{P(\theta_n|EMG)P(EMG)}{\sum_{EMG} P(\theta_n|EMG)P(EMG)}$$

Bayesian Reconstruction



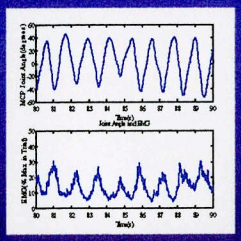
Extension

Flexion

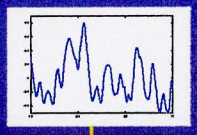


Bayesian Reconstruction

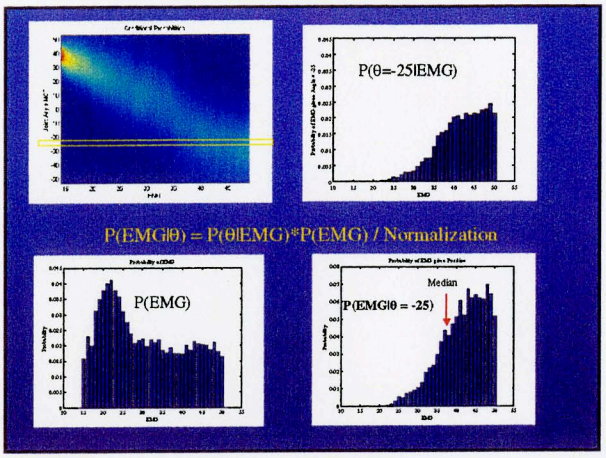
Training Data

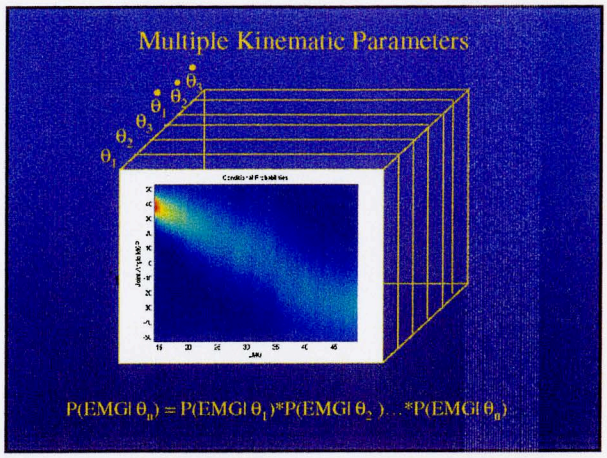


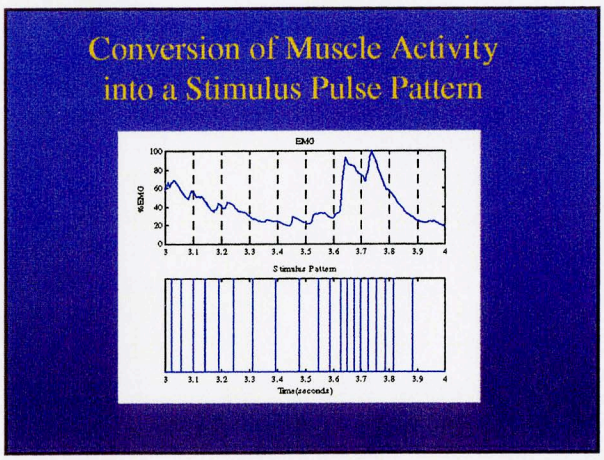
Desired Joint Angle

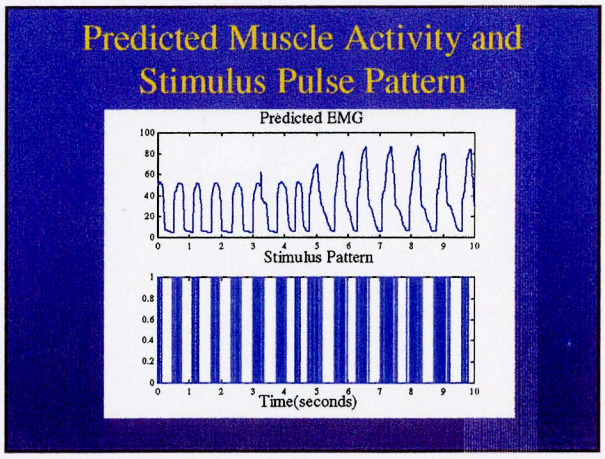


Predicted Muscle Activity

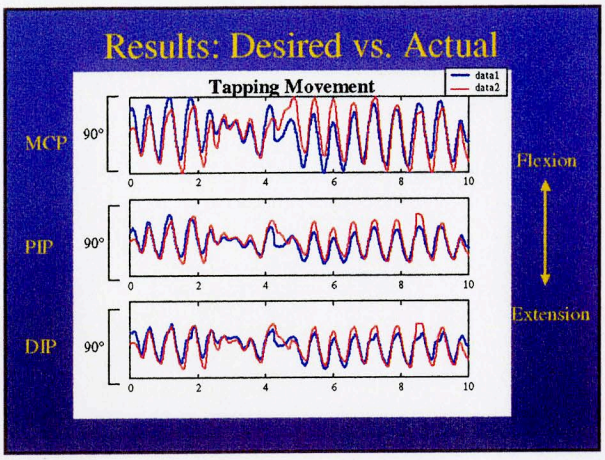

P(θ=-25|EMG)
P(EMG|θ) = P(θ|EMG)*P(EMG) / Normalization
P(EMG)
Median
P(EMG|θ = -25)


Multiple Kinematic Parameters
P(EMG| θn) = P(EMG| θ1)*P(EMG| θ2)…*P(EMG| θn)

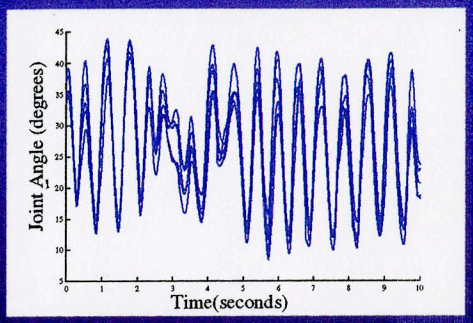

Conversion of Muscle Activity into a Stimulus Pulse Pattern
EMG
%EMG
Stimulus Pattern
Time(seconds)


Predicted Muscle Activity and Stimulus Pulse Pattern
Predicted EMG
Stimulus Pattern
Time(seconds)

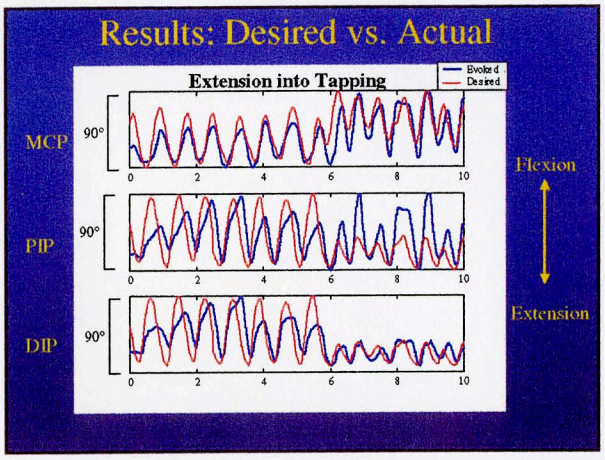
Results: Desired vs. Actual



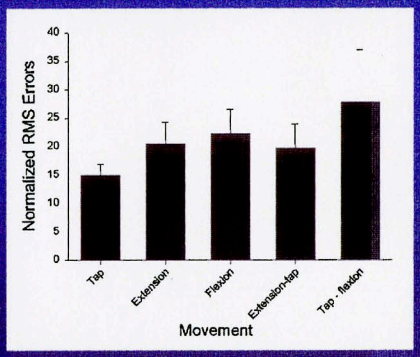
Reproducibility of Movements



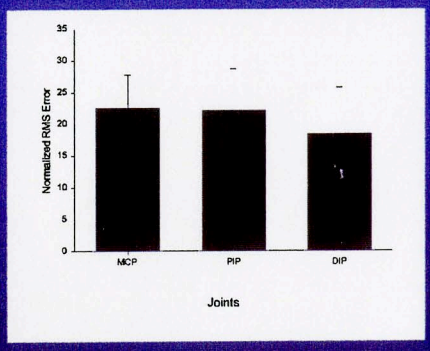
Results: Desired vs. Actual



Results: Errors Across Movements



Results: Errors Across Joints



Conclusions

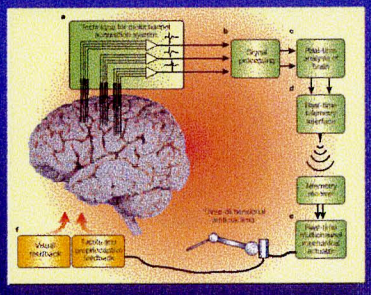
- 10-25% error for different movements
- 18-25% error for different joints

•This reasonable correspondence between desired and evoked movements indicates that this approach might provide a flexible means to control FES systems and thereby expand the repertoire of motor functions available to paralyzed individuals.

Future Directions

- Apply method to the four fingers and thumb
- Apply method to the shoulder and elbow
- Use in combination with a neural prostheses

Neural Prosthetic Device



M. Nicolelis *Nature* January 2001

Appendix F

Digitized Video Clips of Restored Movement